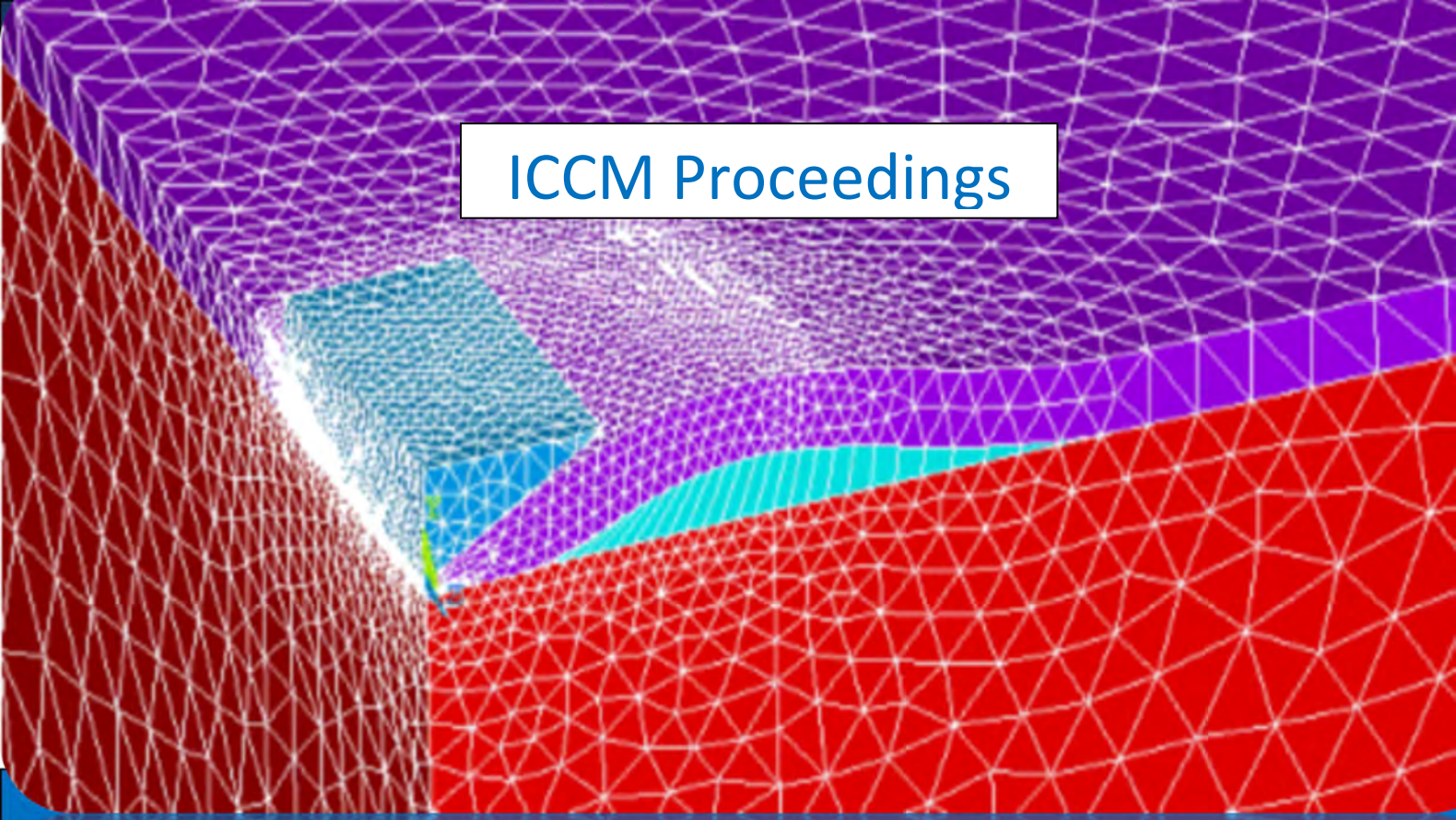


ICCM Proceedings

Proceedings of the International Conference on Computational Methods

(Vol. 11, 2024)

15th ICCM, 16th-18th July 2024

Editors: **G. R. Liu, Vu-Hieu Nguyen**

ScienTech Publisher

ISSN 2374-3948 (online)

ICCM2024

Proceedings of the International Conference on Computational Methods (Vol. 11, 2024)

15th ICCM, 15th-18th July 2024, Virtual Conference

Edited by

Guirong Liu
University of Cincinnati, USA

Vu-Hieu Nguyen
University of Paris-Est Créteil, France

Proceedings of the International Conference on Computational Methods, Vol.11, 2024

This volume contains full papers accepted by the 15th ICCM, 15th-18th July 2024.

First Edition, July 2024

International Standard Serial Number: ISSN 2374-3948 (online)

Papers in this Proceedings may be identically cited in the following manner: Author names, *Paper title, Proceedings at the 15th ICCM2024, 15th-18th July 2024, online*, Eds: Guirong Liu, Vu-Hieu Nguyen, ScienTech Publisher.

Published by
Scientech Publisher LLC, USA
https://www.sci-en-tech.com/

WELCOME MESSAGE

Dear Colleagues and Friends,

It is an honor to welcome you to the 15th International Conference on Computational Methods (ICCM2024) from July 15th to July 18th, by the University of Paris-Est Créteil. Although it was planned to be held in Paris, this year edition will run as virtual conference in order to avoid traveling difficulties to France during the Olympics Games period.

From the first edition organized by professor GR Liu in Singapore in 2014, the ICCM conference series have a long history of gathering the researchers who are in the field of computational science and engineering to exchange ideas and to strengthen international academic networks. The ICCM conference now is a well-established forum giving real opportunity to discuss new issues, tackle complex problems and find advanced enabling solutions able to shape new trends in Computational Science.

As previous years, the conference program accommodate presentations on a wide range of topics related to computational technology and we are very pleased to receive wide-spectrum contributions by experts and researchers from Asia, Europe, America and worldwide. All contributions have been peer-reviewed and the accepted presentations, including abstracts and papers, will be published on our conference website.

We are thankful to the plenary speakers, the conference organizing committee members, the session chairs, and additional reviewers who have made this conference a success. Most of all, we thank you, the participants, for enriching the conference by your presence. We hope you will enjoy the content, get new ideas and have a good time.

We are looking forward to your participation and continued engagement for the future ICCM conferences.

Professor Vu-Hieu Nguyen
Conference Chairman
University of Paris-Est Créteil
France

Professor Guirong Liu
Honorary Conference Chairman
University of Cincinnati
USA

ORGANIZATION COMMITTEES

Conference Chairman
Vu-Hieu Nguyen, Universite Paris-Est Creteil, France

Honorary Chairman
Guirong Liu, University of Cincinnati, United States

France Organization Committee
Boumediene Nedjar, Université d'Évry Paris-Saclay, France
Cuong Ha-Minh, École normale supérieure Paris-Saclay, France
Fakhraddin Seyfaddini, Université Clermont Auvergne, France
Sara Touhami, Université Paris-Est Créteil, France
Felipe Figueredo Rocha, Université Paris-Est Créteil, France
Van-Thanh Nguyen, Université Paris-Est Créteil, France

International Scientific Advisory Committee (Sorting by last name)

Addessi Daniela (Italy)
Atroshchenko Elena (Australia)
Birken Philipp (Sweden)
Bui Ha (Australia)
Bui Tinh Quoc (Japan)
Chen Haibo (China)
Chen Lei (USA)
Chen Shaohua (China)
Chen Weiqiu (China)
Chen Zhen (USA)
Chiu Wingkong (Australia)
Cui Fangsen (Singapore)
Das Raj (Australia)
Dong Leiting (China)
Dong Xiangwei (China)
Feng Chuang (China)
Feng Dianlei (China)
Fu Zhuojia (China)
Gao Wei (Australia)
Gao Xiao-Wei (China)
Gravenkamp Hauke (Germany)
Gu Yuantong (Australia)
Huang Yu (China)
Jiang Chao (China)
Jin Feng (China)
Kang Zhan (China)
Lee Chin-Long (New Zealand)
Lee Ik-Jin (South Korea)
Li Bing (China)
Li Chenfeng (UK)
Li Eric (UK)
Li Hua (Singapore)
Liang Lihong (China)
Liu Bin (China)
Liu Moubin (China)
Liu Xiaojing (China)
Liu Yan (China)
Mahmood Jabareen (Israel)
Miller Karol (Australia)
Montenegro Rafael (Spain)
Natarajan Sundararajan (India)
Nguyen The Hung (Vietnam)
Ni Bingyu (China)
Nithiarasu Parumal (UK)
Ogino Masao (Japan)
Onishi Yuki (Japan)
Orlande Helcio (Brazil)
Papadrakakis Manolis (Greece)
Peng Qing (China)
Ponthot Jean-Philippe (Belgium)
Quek Siusin Jerry (Singapore)
Reali Alessandro (Italy)
Reddy Daya (South Africa)
Saitoh Takahiro (Japan)
Shen Lian (USA)
Shen Yongxing (China)
Shioya Ryuji (Japan)
Son Gihun (South Korea)
Song Chongmin (Australia)
Stefanou George (Greece)
Su Cheng (China)
Suo Tao (China)
Tadano Yuichi (Japan)
Tan B.C. Vincent (Singapore)
Tian Rong (China)
Tian Zhao-Feng (Australia)
Trung Thoi Nguyen (Vietnam)
Tsubota Ken-Ichi (Japan)
Wan Decheng (China)
Wang Bo (China)
Wang Cheng (China)
Wang Dongdong (China)
Wang Hu (China)
Wang Jie (China)
Wang Jizeng (China)
Wang Lifeng (China)
Wang Lihua (China)
Wang Lipo (China)
Wang Yue-Sheng (China)
Wu Wei (Austria)
Xiao Feng (Japan)
Xiao Jinyou (China)
Xu Fei (China)
Yang Chun Charles (Singapore)
Yang Judy (Taiwan)
Yang Qingcheng (China)
Yang Zhenjun (China)
Yao Jianyao (China)
Ye Hongling (China)
Ye Qi (China)
Yeo Jingjie (USA)
Youn Sung-Kie (South Korea)
Yu Chengxiang Rena (Spain)
Zhan Haifei (Australia)
Zhang Aman (China)
Zhang Chuanzeng (Germany)
Zhang Guiyong (China)
Zhang Jian (China)
Zhang Lucy (USA)
Zhang Xiong (China)
Zhang Zhe (China)
Zhang Zhong (China)
Zhong Zheng (China)
Zhou Annan (Australia)
Zhou Kun (Singapore)
Zhuang Zhuo (China)

TABLE OF CONTENTS

The crush behavior of glass fiber reinforced polymer composite pipe fabricated by pullwinding process

Zhangxing Chen[1,2], Jia Li[1], Zhiqiang Cao[1], †*Haibin Tang[1]

[1]Nanjing University of Science and Technology, Nanjing, 210094, China

[2]Key Laboratory of Composite Materials on Electrical Insulation, National Energy Administration, Jiangsu Shemar Electric CO., LTD., Nantong, 226017, China

*Presenting author: htang28@njust.edu.cn

†Corresponding author: htang28@njust.edu.cn

Abstract

The fiber reinforced polymer composites are gradually becoming the primary promotional product for electric power companies. In the present work, the crush behavior of glass fiber reinforced polymer composite pipes fabricated by pullwinding process is analyzed. The impact test of composite pullwinding pipe is conducted with large-scale cranes. The loading directions are 0, 30, and 90 degrees, and different impact energies including 66J, 133J, and 266J are compared. Further, finite element models are established in Ls-Dyna. The crush failure behaviors of glass fiber reinforced polymer composite pullwinding pipes are reproduced.

Keywords: Crash behavior, glass fiber reinforced polymer composite, pullwinding process

1. Introduction

Due to its excellent axial mechanical properties and high material utilization during the production process, fiber reinforced polymer composites are gradually becoming the primary promotional product for electric power companies [1-4]. However, these fiber reinforced polymer composites are weak along the transverse and out-of-plane directions. The commercial glass fiber reinforced polymer composite pipes are easily broken caused by bumps, transportation, and falls, which have an impact on their service performance. Thus, the crush failure mechanism of the glass fiber reinforced polymer composite pipes under low-speed impact is meaningful, and plays a significant role for the safety design during the entire transportation.

The primary factors influencing the impact performance of fiber reinforced polymer composite are categorized into three main categories: material architecture, impact load characteristics, and external environmental factors [5-8]. Among these, the inherent properties of the composite material architecture play a crucial role in determining its impact resistance. This architecture includes aspects such as fiber material, matrix composition, and interfacial properties between the fibers and the matrix. During the low-speed impact process, fiber

reinforced polymer composite materials exhibit various mechanisms for absorbing impact energy, including matrix cracking, fiber pull-out, fracture, and delamination. To enhance the impact resistance, improvement can be applied through resin formulation, hybridization with multiple types of fibers, 3D fiber weaving, as well as the construction of energy-absorbing structures. Currently, there are few investigations about the impact failure of glass fiber reinforced polymer composite pipes fabricated by pullwinding process.

In this present work, the failure mechanisms and performance degradation under different loading conditions are investigated for glass fiber reinforced polymer composite pipes fabricated by pullwinding process. The finite element models are established in commercial software Ls-dyna to evaluate the progressive failure behavior of these composite pullwinding pipes.

2. Impact test of glass fiber reinforced polymer composite pullwinding pipes

The glass fiber reinforced polymer composite pullwinding pipes are provided by Jiangsu Shemar Electric Company. The pipes are composed by three plies, and the orientation angles are as follows: 0/90/0, the thickness of which are 1.5 mm, 0.5 mm, and 4 mm. The internal and external diameters of the pullwinding pipes are 198 mm and 212 mm, respectively. The impact test of composite pullwinding pipe is conducted with large-scale cranes as shown in Figure 1, and the weight of the impactor is around 66 Kg. The impact energies are set as 66J, 133J, and 266J, respectively. The loading directions are 0, 30, and 90 degrees, respectively. The 0 degree is along the pipe direction, while 90 degree is perpendicular to the pipe direction. The detailed geometries and quantity of the specimens are listed in Table 1.

To suppress the bouncing or intense vibration caused by the impact loading, a steel stoper is used during the 0 degree test. In addition, a fixture with a 30 degree slope is adopted during the 30 degree test. the composite pullwinding pipe is fixed on the slope, and also the steel stoper is used to suppress the sliding. During the 90 degree test, the length of the composite pullwinding pipe is much larger than other cases to match the size of the impactor.

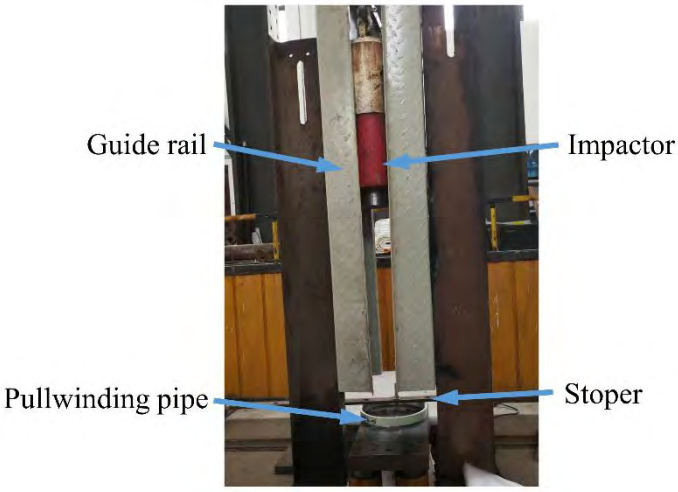


Figure 1. The setup of the impact test for composite pullwinding pipes

Table 1. The test plan for composite pullwinding pipes

Impact energy (J)	Loading direction	Length of specimen (mm)	Quantity
66	0°	25	5
	30°		5
	90°	250	5
133	0°	25	5
	30°		5
	90°	250	5
266	0°	25	5
	30°		5
	90°	250	5

3. Experimental results of composite pullwinding pipes under impact loading

3.1. Results of the impact test along the 0 degree direction

When the loading direction is 0°, i.e., the pipe direction, the different damage behaviors of the composite pullwinding pipes are observed with different impact energies at the cross-section of the specimens, as shown in Figure 2(a)-(c). From the graph, it is obvious that the destructions of the composite pullwinding pipes become more pronounced as the impact energy increases. When the impact energy is 66J, the broken at the cross-section of the composite pullwinding pipe is limited. Nevertheless, there are some minor indentations near the impacted edge due to shear force from the impactor. When the impact energy is 133J, the cross-section damage of the composite pullwinding pipe is close to that of 66J. The indentations near the impact edge tends more distinct. When the impact energy is 266J, noticeable squash occurs at both the impact center and the impact edge of the composite pullwinding pipe. Along the thickness direction, a crack forms that penetrates from the outer surface to the inner surface at the impact point.

Not only the cross-section damages exhibit different forms, but the lateral surface of the composite pullwinding pipes also follows a similar trend, as depicted in Figure 2(d)-(f). Observations reveal the varying degrees of damage to the lateral outer surface of the composite pullwinding pipes as the energy levels increase. When the impact energy is 66J, there are no significant cracks on the lateral surface, and the overall integrity remains intact. When the impact energy increases to 133J, A minor surface crack appears at the impact edge. When the impact energy reaches 266J, the destruction becomes more pronounced. Both the vertical cracks penetrate from the upper to the bottom.

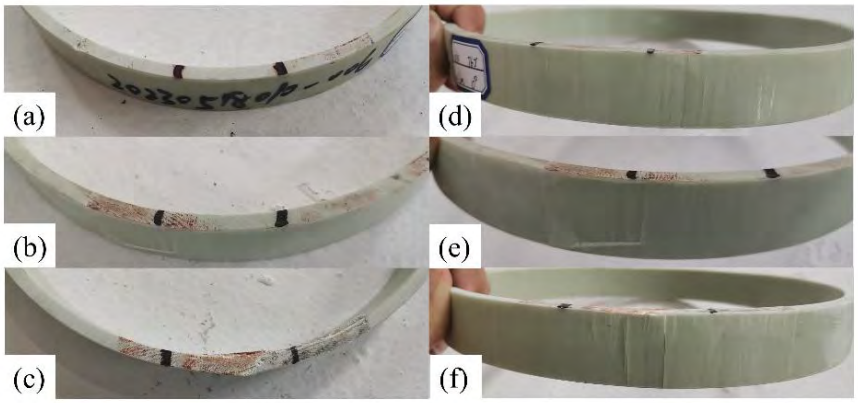


Figure 2. The crush behavior of composite pullwinding pipes under 0 degree impact loading for (a) and (d) the impact energy is 66J, (b) and (e) the impact energy is 133J, and (c) and (f) the impact energy is 266J.

3.2. Results of the impact test along the 30 degree direction

As shown in Figure 3(a)-(c), the crash failure of inter and outer at the cross-section is different for glass fiber reinforced polymer composite pullwinding pipes under 30 degree impact loading. The damage is closer to the outer surface at the cross-section owing to the contact region with the impact. As the energy level increases, the extent of deformation from this compression also grows. Since the impact center cannot be accurately setup, the failure deformation also varies under the same impact loading. In general, when the impact energy is 66J, there is minimal change at the lateral surface. When the impact energy is 133J, the lateral surface experiences superficial peeling, and the internal fibers suffer partial damage. However, the main structure of the composite pullwinding pipes remains uncracked. When the impact energy is 266J, the lateral surface peeling becomes more pronounced, with some cracks extending from the impact center to both the inner and outer surfaces. The edge compression is also more evident. These observations highlight the distinct lateral surface damage modes as impact energy continues to rise.

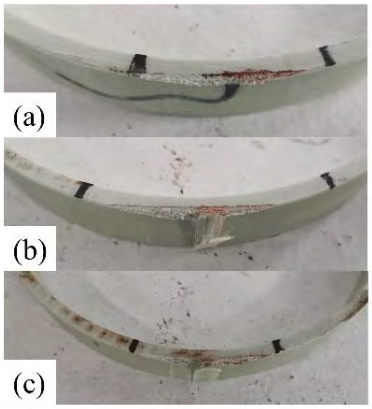


Figure 3. The crush behavior of composite pullwinding pipes under 30 degree impact loading for (a) the impact energy is 66J, (b) the impact energy is 133J, and (c) the impact energy is 266J.

3.3. Results of the impact test along the 90 degree direction

As shown in Figure 4(a)-(c), the crash deformation of glass fiber reinforced polymer composite pullwinding pipes under 90 degree impact loading is similar for different loading levels. Two cracks appear across the whole pipes distributed on the two sides of the outer for the composite pullwinding pipes, while the impact energy ranges from 66J to 266J. By comparison, the composite pullwinding pipes maintain their circular shape at 66J and 133J. However, when the impact energy is 266J, there is a slight reduction in circularity. In addition, owing to the force from the impactor during impact loading, some cracks are also found at the internal wall of the contact region for the composite pullwinding pipes, as given in Figure 4(d)-(f).

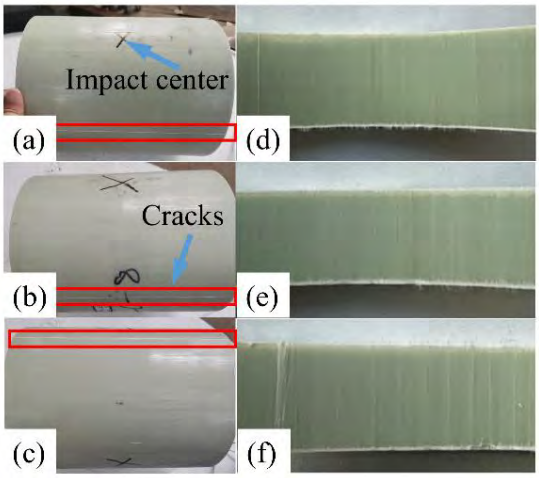


Figure 4. The crush behavior of composite pullwinding pipes under 90 degree impact loading for (a) and (d) the impact energy is 66J, (b) and (e) the impact energy is 133J, and (c) and (f) the impact energy is 266J.

4. Numerical simulation of composite pullwinding pipes under impact loading

4.1. Numerical simulation of the impact test along the 0 degree direction

In order to better understand the crush failure behaviors of glass fiber reinforced polymer composite pullwinding pipes, the finite element models are established in the current work. The dimensions of the impactor, stoper, and specimens are from the experiments. The steel impactor and stoper are set as rigid bodies, and the rigid wall is added at the bottom. Tshell element is used for the composite pullwinding pipe, and the *MAT_ ENHANCED_COMPOSITE_DAMAGE (54/55) material card is adopted for plies of glass fiber reinforced composite pullwinding pipes. To model the contacts between the steel impactor and the pipe, the *CONTACT_AUTOMATIC_SURFACE_TO_SURFACE is selected. In order to reduce the time during calculation, the height of the impactor is set to be different from the tests, which is very close to the composite pullwinding pipes, and then an initial speed is given to the impactor. The impact energy is the same for both experiments and

simulations. When the impact energy is 66J, 133J, and 266J, the initial speed is 1.4 m/s, 2.03 m/s, and 2.836 m/s, respectively.

The finite element model of impact test along the 0 degree direction is given in Figure 5(a), and the numerical crush deformations of composite pullwinding pipes under 66J, 133J, and 266J are shown in Figure 5(b)-(d). Due the effect of impact loading, the upper around the impact region is failed for composite pullwinding pipe when the impact energy is 66J. In addition, with the increment of the impact energy, the failure tends remarkable, and the impact region is fully broken when the impact energy is 266J, which are consistent with the experimental results.

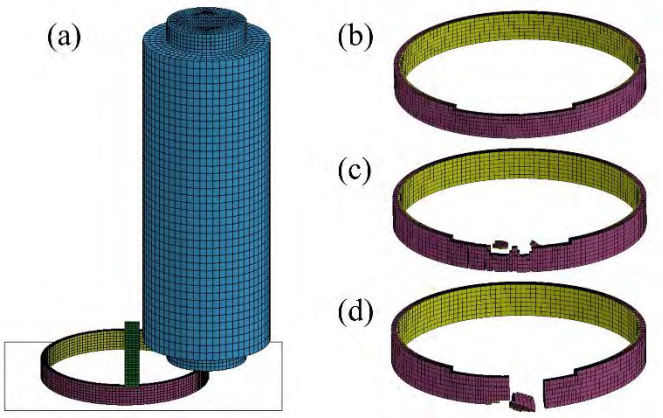


Figure 5. Numerical simulation of the impact test along the 0 degree direction for composite pullwinding pipes for (a) the finite element model, (b) the impact energy is 66J, (c) the impact energy is 133J, and (d) the impact energy is 266J.

4.2. Numerical simulation of the impact test along the 30 degree direction

The simulation of the 30 degree impact loading is similar to that of the 0 degree impact loading. The composite pullwinding pipe, steel stoper, as well as rigid wall are all rotated by 30° to simulate the experimental scenario. To prevent mutual interference between the stoper and the rigid wall, their interactions are disabled in the simulation settings. An appropriate friction coefficient between the stoper and the composite pullwinding pipe is applied to match the loose condition of the stoper during the 30 degree impact loading.

The finite element model of impact test along the 30 degree direction is given in Figure 6(a), and the numerical crush deformations of composite pullwinding pipes under 66J, 133J, and 266J are shown in Figure 6(b)-(d). Overall, the simulation results align reasonably well with the experimental observations. The simulation results clearly indicate that under a 30° impact, the damage to the composite pullwinding pipe is concentrated near the impact point, which is consistent with the experimental crush behavior. As the constrain of the composite pullwinding pipe is relatively loosely fixed on the slope during experiments, the inevitable sliding is not reproduced in the numerical simulation. Then, the numerical failure of the

composite pullwinding pipe is slightly obvious compared with the experiments for each loading level.

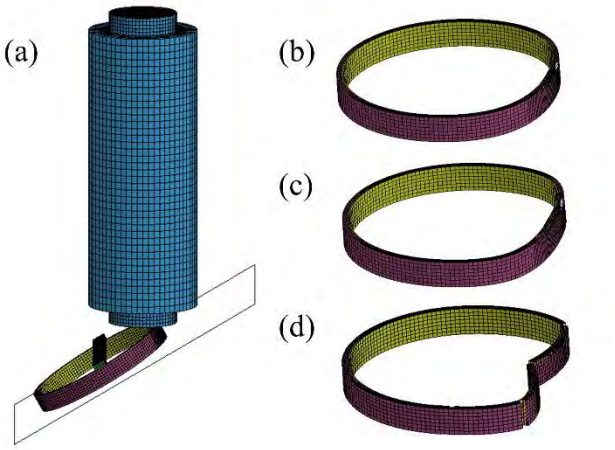


Figure 6. Numerical simulation of the impact test along the 30 degree direction for composite pullwinding pipes for (a) the finite element model, (b) the impact energy is 66J, (c) the impact energy is 133J, and (d) the impact energy is 266J.

4.3. Numerical simulation of the impact test along the 90 degree direction

The simulation for a 90° impact loading is also similar in most aspects. As the Tshell element is not suited to simulate the rebound process of the composite pullwinding pipe when subjected to circumferential impact. The shell element is used to construct the model. The ply orientation angles of the composite pullwinding pipe remain set at 0/90/0, and 3, 2, and 4 integration points are used in the three plies.

The finite element model of impact test along the 90 degree direction is given in Figure 7(a). Due to the limitations of the shell element, the failure process of individual element is not displayed. Thus, the image of the history variable #3 from the post-processing software Ls_PrePost is used to show the failure behavior, as given in Figure 7(b)-(d). Similar to the damage observed during the 90° impact experiment, the simulated element failures occur primarily on the inner wall at the top region and the two sides of the outer wall for the composite pullwinding pipe. As the energy level increases, the maximum compression distance also rises, while circularity gradually decreases. Additionally, the simulation effectively captures the rebound phenomenon generated during impact, overall validating the experimental process.

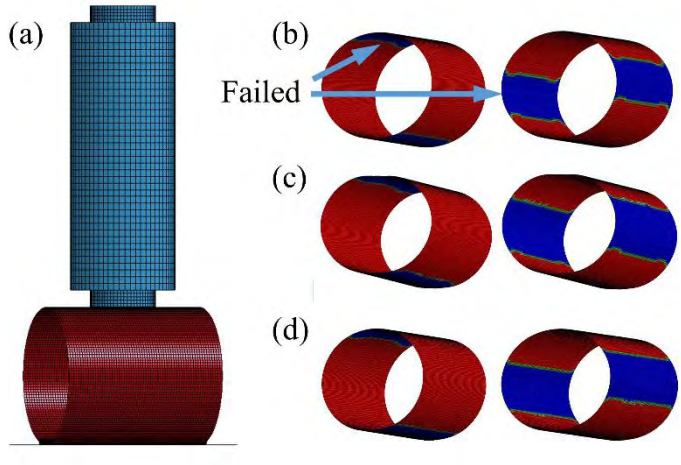


Figure 7. Numerical simulation of the impact test along the 90 degree direction for composite pullwinding pipes for (a) the finite element model, (b) the impact energy is 66J, (c) the impact energy is 133J, and (d) the impact energy is 266J.

5. Conclusions

In the present work, the crush behavior of glass fiber reinforced polymer composite pullwinding pipes is investigated. The loading directions are 0, 30, and 90 degrees, and different impact energies including 66J, 133J, and 266J are compared. When the loading direction is 0 degree, the damage primarily concentrates at the edge of contact region of the impactor and the composite pullwinding pipe. When the loading direction is 30 degree, the destruction occurs near the center of the contact region. When the loading direction is 90 degree, the failure manifests on the outer wall of the pipe on both sides and the inner wall of the pipe at the impact point. With the increment of impact energy, the damage tends obvious. Using numerical modeling and simulation with Ls-Dyna, we can effectively replicate the failure modes of the composite pullwinding pipe under 0, 30, and 90 degrees impact loading.

Acknowledgement

This work is supported by National Natural Science Foundation of China under Grant Number 52305382, Jiangsu Shemar Electric CO., LTD. under Grant Number 1234021280931, and State Key Laboratory of Mechanics and Control for Aerospace Structures (Nanjing University of Aeronautics and astronautics) under Grant Number MCMS-E-0423Y01. The authors also gratefully acknowledge the contribution of Prof. Guowei Zhou from Shanghai Jiao Tong University and Dr. Li Yang for the detailed discussion of the simulation work.

Reference

[1] Tang, H., Zhou, G., Sun, Q., Avinesh, O., Meng, Z., Engler-Pinto, C., Su, X. (2021) Experimental and computational analysis of bending fatigue failure in chopped carbon fiber chip reinforced composites, *Composite Structures* **275**, 114402.

[2] Tang, H., Chen, H., Sun, Q., Chen, Z., Yan, W. (2021) Experimental and computational analysis of structure-property relationship in carbon fiber reinforced polymer composites fabricated by selective laser sintering,

Composites Part B: Engineering **204**, 108499.
[3] Tang, H., Sun, Q., Li, Z., Su, X., Yan, W. (2021) Longitudinal compression failure of 3D printed continuous carbon fiber reinforced composites: An experimental and computational study, *Composites Part A: Applied Science and Manufacturing* **146**, 106416.
[4] Zhang, S., Tang, H., Tang, D., Liu, T., Liao, W. (2024) Effect of fabrication process on the microstructure and mechanical performance of carbon fiber reinforced PEEK composites via selective laser sintering, *Composites Science and Technology* 246, 110396.
[5] Striewe, J., Reuter, C., Sauerland, K.H., Tröster, T. (2018) Manufacturing and crashworthiness of fabric-reinforced thermoplastic composites, *Thin-Walled Structures* **123**, 501-508.
[6] Liu, Q., Ou, Z., Mo, Z., Li, Q., Qu, D. (2015) Experimental investigation into dynamic axial impact responses of double hat shaped CFRP tubes, *Composites Part B: Engineering* **79**, 494-504.
[7] Wang, Y., Feng, J., Wu, J., Hu, D. (2016) Effects of fiber orientation and wall thickness on energy absorption characteristics of carbon-reinforced composite tubes under different loading conditions, *Composite Structures* **153**, 356-368.
[8] Zhou, G., Tang, H., Sun, Q., Li, D., Peng, Y., Zeng, D., Su, X. (2021) Analysis of the crushing behaviors of woven carbon fiber reinforced plastic hat section component under dynamic bending and axial crushing loading, *Thin-Walled Structures* **161**, 107426.

A novel research on the base force element method of complementary energy principle for finite strain problems

***Zhonghai Li, and †Yijiang Peng**

Key Laboratory of Urban Security and Disaster Engineering Ministry of Education, Beijing University of Technology, Beijing, P. R. China

*Presenting author: lizhonghai@emails.bjut.edu.cn
†Corresponding author: pengyijiang@bjut.edu.cn

Abstract

The finite element method based on the complementary energy principle and its combination with artificial neural networks is a worthwhile research topic. A new Base Force Element Method (BFEM) model is proposed in this study, which can solve the problems of large elastic deformation and finite strain in incompressible hyperelastic materials. The complementary energy function of nonlinear elasticity has not yet been proposed. The complementary energy function is replaced by a pre-trained Back-Propagation (BP) neural network to establish the constitutive relationship. This makes it easy for finite strain problems to be studied by the BFEM model. When the constitutive relationships of different incompressible hyperelastic materials were described, only two parameters needed to be changed. The governing equations of the BFEM model are general and concise, and its derivation process does not involve an approximate representation of shape functions. The calculation results of the examples indicate that the model has high accuracy. This new numerical method fills the gap in using the complementary energy finite element method to calculate the finite strain problem of incompressible hyperelastic materials.

Keywords: Complementary energy principle, BFEM, large elastic deformation, finite strain, BP neural network, solid element

Introduction

When the nonlinear complementary energy variational principle is established by the second type of Piola-Kirchhoff stress tensor, a coupling term containing force and displacement appears in its integral term [1]. This makes it difficult for numerical methods based on the complementary energy principle to solve geometric nonlinear problems. This difficulty is overcome by the base force [2]. Traditional stress tensors can be replaced by base force to describe the stress state at a certain point [3]. The coupling term is eliminated in the process of establishing the complementary energy principle using base force. According to the complementary energy principle of base force, a new finite element method - BFEM is proposed [2]. This is a new way of calculating nonlinear solid mechanics. Recently, BFEM has been used to accurately solve the large elastic deformation problems of trusses and analyze the stability of beams [4, 5]. The constitutive relationship used in BFEM is still linear and elastic, the research problems are still limited to the range of linear elastic materials.

Many scholars have proposed strain energy functions to characterize the constitutive relationships of nonlinear elasticity [6-8]. However, the complementary energy function of nonlinear elasticity has not yet been proposed. This makes it difficult to solve finite strain problems using complementary energy finite element methods. The strain energy functions are established based on the phenomenological method [9-11]. These functions are mostly represented by strain invariants or principal tensile ratios. Similarly, can the complementary

energy function be represented by stress invariants or principal stress? The answer is yes when the constitutive relationship is linear elastic. The linear complementary energy function is obtained using the Legendre transformation. If the constitutive relationship is nonlinear elastic, it is difficult to obtain the corresponding complementary energy function using the Legendre transformation. This is because the constitutive relationship must be inverted during the derivation of the complementary energy function, but the nonlinear constitutive relationship may not be reversible [12, 13]. To avoid this difficulty, complementary energy functions can be established through phenomenological methods. However, if the material is replaced, it is necessary to fit and analyze the sample again, so the generality of this method is poor. Neural networks have excellent data feature extraction capabilities [14-16], reducing the cost of manual processing of samples. When the material is replaced, the neural network only needs to be retrained based on different samples. Therefore, using constitutive relationships formed by neural networks to establish the complementary energy principle of finite strain is a general method.

To solve the finite strain problems of incompressible hyperelastic materials using complementary finite element models, a BFEM based on constitutive relationships established by BP neural networks is proposed. The premise of the complementary energy function obtained by Legendre transformation is that the nonlinear constitutive relationship is reversible. By using the BP neural network, the strain-stress relationship can be directly obtained, avoiding the use of the Legendre transform. To provide training data to the BP neural network, the test samples were inverted. The weights and biases are extracted from the pre-trained BP neural network and substituted into the mathematical expression of the BP neural network. The two unknown parameters in the element compliance matrix can be predicted by this expression. In this way, the governing equations are complete and the finite strain problems of incompressible materials can be solved.

BFEM model

Assuming that $\boldsymbol{\sigma}$ and $\boldsymbol{\tau}$ are the Cauchy stress tensor and Piola stress tensor, respectively. These stress tensors can be represented by the base force $\boldsymbol{T}^i$ as follows [2]:

$$\boldsymbol{\sigma} = \frac{1}{V_Q}\boldsymbol{T}^i \otimes \boldsymbol{Q}_i \,, \tag{1}$$

$$\boldsymbol{\tau} = \frac{1}{V_P}\boldsymbol{T}^i \otimes \boldsymbol{P}_i \,, \tag{2}$$

where $\boldsymbol{P}_i$ and $\boldsymbol{Q}_i$ are the covariant vector bases of the initial configuration and the deformed configuration, respectively, V_P and V_Q are the base volumes before and after deformation. Assuming that the volume of an elastic body is represented by V, its boundary S can be divided into Dirichlet boundary S_u and Neumann boundary S_σ. Both boundaries satisfy the conditions $S_u \cup S_\sigma = S$ and $S_u \cap S_\sigma = \varnothing$. $\boldsymbol{m}$ is the unit normal vector on boundary S_u and $\bar{\boldsymbol{u}}$ is the given displacement. The complementary energy principle of base force [17] is

$$\int_V W_c\left(\boldsymbol{T}^i\right)\mathrm{d}V - \int_{S_u}\frac{m_i}{V_P}\bar{\boldsymbol{u}}\cdot\boldsymbol{T}^i\mathrm{d}S \,, \tag{3}$$

where $m_i = \boldsymbol{P}_i \cdot \boldsymbol{m}$, and W_c is the complementary energy function. Since $\boldsymbol{T}^i$ and displacement gradients $\boldsymbol{u}_i$ are conjugated,

$$\frac{1}{V_P}\boldsymbol{T}^i \cdot \boldsymbol{u}_i = W_c + W \,, \tag{4}$$

where W is the strain energy function. Similar to the polar decomposition of deformation gradients, $\boldsymbol{u}_i$ can be divided into [18]

$$\boldsymbol{u}_i = \boldsymbol{u}_{id} + \boldsymbol{u}_{ir}, \tag{5}$$

where $\boldsymbol{u}_{id}$ is the deformation displacement gradients, and $\boldsymbol{u}_{ir}$ is the rotation displacement gradients. Therefore, formula (4) is divided into

$$W_{cd} = \frac{1}{V_P}\boldsymbol{T}^i \cdot \boldsymbol{u}_{id} - W, \quad W_{cr} = \frac{1}{V_P}\boldsymbol{T}^i \cdot \boldsymbol{u}_{ir}, \tag{6}$$

where W_{cd} is the deformed part of W_c, and W_{cr} is the rotating part of W_c. The complementary energy principle of large elastic deformation represented by $\boldsymbol{T}^i$ is obtained by substituting formula (6) into formula (3). It is worth noting that W_{cd} should be a specific expression rather than the form presented by formula $(6)_1$. For example, the W_{cd} of linear elastic materials can be expressed by the first invariant J_1 and the second invariant J_2 of the stress tensor $\boldsymbol{\sigma}$:

$$W_{cd}\left(\boldsymbol{T}^i, \theta\right) = \frac{1+\nu}{2E}\left(J_2 - \frac{\nu}{1+\nu}J_1^{\,2}\right), \tag{7}$$

where θ is the rotation angle, ν is the Poisson's ratio, and E is the Young's modulus. Let us assume that the W_{cd} of incompressible hyperelastic materials is

$$W_{cd} = W_{cd}\left(J_{1\tau}, J_{2\tau}\right), \tag{8}$$

where $J_{1\tau}$ and $J_{2\tau}$ are the first and second invariants of stress $\boldsymbol{\tau}$, respectively. The complementary energy of the element is

$$W_{cd}^e = \int_V W_{cd}\left(J_{1\tau}, J_{2\tau}\right)\mathrm{d}V. \tag{9}$$

The average Piola stress of the element is

$$\overline{\boldsymbol{\tau}} = \frac{1}{V}\int_V \boldsymbol{\tau}\mathrm{d}V = \frac{1}{V}\int_V \frac{1}{V_P}\boldsymbol{T}^i \otimes \boldsymbol{N}_i \mathrm{d}V, \tag{10}$$

where $\boldsymbol{N}_i$ is the covariant vector base of the intermediate configuration after rotation, $\boldsymbol{N}_i = \boldsymbol{P}_i + \boldsymbol{u}_{ir}$. According to the Gaussian theorem, formula (10) is rewritten as

$$\overline{\boldsymbol{\tau}} = \frac{1}{V}\int_A \boldsymbol{T} \otimes \boldsymbol{N}\mathrm{d}S, \tag{11}$$

where $\boldsymbol{T}$ is the stress vector acting on boundary A, and $\boldsymbol{N}$ is the intermediate position vector of the $\boldsymbol{T}$ action point. If the element is small enough, it can be approximately assumed that the stress is evenly distributed on each surface. The area integral in formula (11) can be eliminated:

$$\overline{\boldsymbol{\tau}} = \frac{1}{V}\boldsymbol{T}^\alpha \otimes \boldsymbol{N}_\alpha, \tag{12}$$

where α is the label of the surface of the element, and $\boldsymbol{N}_\alpha = \boldsymbol{R}\cdot\boldsymbol{P}_\alpha$ is the intermediate position vector of the midpoint in surface α. Formula (12) satisfies the Einstein summation convention. According to formulas (10) and (12),

$$\int_V J_{1\tau}\mathrm{d}V = \int_V \boldsymbol{\tau}\mathrm{d}V : \boldsymbol{I} = V\overline{\boldsymbol{\tau}} : \boldsymbol{I} = \boldsymbol{T}^\alpha \cdot \boldsymbol{N}_\alpha, \tag{13}$$

$$\int_V J_{2\tau}\mathrm{d}V = \int_V \boldsymbol{\tau}\mathrm{d}V : \int_V \boldsymbol{\tau}\mathrm{d}V = V^2\overline{\boldsymbol{\tau}} : \overline{\boldsymbol{\tau}} = \left(\boldsymbol{T}^\alpha \cdot \boldsymbol{T}^\beta\right)\left(\boldsymbol{N}_\alpha \cdot \boldsymbol{N}_\beta\right). \tag{14}$$

Let us assume that the first and second invariants of stress $\overline{\boldsymbol{\tau}}$ are

$$J_{1T} = \overline{\boldsymbol{\tau}} : \boldsymbol{I} = \frac{1}{V}\boldsymbol{T}^\alpha \cdot \boldsymbol{N}_\alpha \tag{15}$$

and

$$J_{2T} = \overline{\boldsymbol{\tau}} : \overline{\boldsymbol{\tau}} = \frac{1}{V^2}\left(\boldsymbol{T}^\alpha \cdot \boldsymbol{T}^\beta\right)\left(\boldsymbol{N}_\alpha \cdot \boldsymbol{N}_\beta\right), \tag{16}$$

respectively. Therefore, the complementary energy form of elements is approximately

$$\int_V W_{cd}\left(J_{1\tau}, J_{2\tau}\right)\mathrm{d}V \approx W_{cd}^e\left(J_{1T}, J_{2T}\right). \tag{17}$$

Similarly,

$$W_{cr}^{e}=\int_{V}W_{cr}\mathrm{d}V=\boldsymbol{T}^{\alpha}\cdot\boldsymbol{u}_{\alpha r}=\boldsymbol{T}^{\alpha}\cdot(\boldsymbol{R}-\boldsymbol{I})\cdot\boldsymbol{P}_{\alpha}\,, \tag{18}$$

where $\boldsymbol{u}_{\alpha r}$ is the rotating part of the displacement of the midpoint in surface α, and $\boldsymbol{R}$ is the rotation tensor. According to the Gaussian theorem, formulas (3), (4), (6), (17), and (18), the complementary function of an element is

$$\pi_{C}=W_{cd}^{e}\left(J_{1T},J_{2T}\right)+W_{cr}^{e}-\overline{\boldsymbol{u}}_{\alpha}\cdot\boldsymbol{T}^{\alpha}\,, \tag{19}$$

where $\overline{\boldsymbol{u}}_{\alpha}$ is the given displacement of the midpoint in surface α. The element surface force $\boldsymbol{T}^{\alpha}$ needs to meet the following constraint conditions:

$$\begin{aligned}&\sum_{\alpha}\boldsymbol{T}^{\alpha}=0 && \text{(Without body force)}\\ &\boldsymbol{T}^{S_{\sigma}}=\overline{\boldsymbol{T}} && \text{(On } S_{\sigma}\text{)}\\ &\boldsymbol{T}^{S_{\alpha}}=\boldsymbol{T}^{S_{\beta}} && \text{(On } S_{\alpha\beta}\text{)}\end{aligned}\,, \tag{20}$$

where $S_{\alpha\beta}$ is the adjacent edge between elements. The Lagrange multiplier method is used to relax the constraints $(20)_1$. The modified complementary functional of elements is obtained:

$$\pi_{c}^{*}\left(\boldsymbol{T},\theta,\boldsymbol{L}\right)=W_{cd}^{e}\left(J_{1T},J_{2T}\right)+W_{cr}^{e}+\boldsymbol{L}\left(\sum_{\alpha}\boldsymbol{T}^{\alpha}\right)-\overline{\boldsymbol{u}}_{\alpha}\cdot\boldsymbol{T}^{\alpha}\,, \tag{21}$$

where $\boldsymbol{L}$ are Lagrange multipliers. The constraints $(20)_2$ and $(20)_3$ can be satisfied in programming. If an elastic body is composed of n elements, the modified complementary energy function is

$$\Pi_{c}^{*}=\sum_{n}\left[\pi_{c}^{*}\left(\boldsymbol{T},\theta,\boldsymbol{L}\right)\right]. \tag{22}$$

The stationary value is taken for formula (21) before assembly. The finite element governing equations are

$$\begin{cases}\dfrac{\partial\pi_{c}^{*}}{\partial\boldsymbol{T}^{\alpha}}=\boldsymbol{C}_{\alpha\beta}\cdot\boldsymbol{T}^{\beta}+(\boldsymbol{R}-\boldsymbol{I})\cdot\boldsymbol{P}_{\alpha}+\boldsymbol{L}-\overline{\boldsymbol{u}}_{\alpha}\\ \dfrac{\partial\pi_{c}^{*}}{\partial\theta}=\dfrac{\partial W_{cd}^{e}}{\partial\theta}+\boldsymbol{T}^{\alpha}\cdot\dfrac{\mathrm{d}\boldsymbol{R}}{\mathrm{d}\theta}\cdot\boldsymbol{P}_{\alpha}\\ \dfrac{\partial\pi_{c}^{*}}{\partial\boldsymbol{L}}=\sum_{\alpha}\boldsymbol{T}^{\alpha}\end{cases}\,, \tag{23}$$

where $\boldsymbol{C}_{\alpha\beta}$ is the compliance matrix. The complementary energy function, $\boldsymbol{C}_{\alpha\beta}$, and $\partial W_{cd}^{e}/\partial\theta$ of linear elastic materials are specific. Although the complementary energy function of hyperelastic materials is unknown, $\boldsymbol{C}_{\alpha\beta}$ and $\partial W_{cd}^{e}/\partial\theta$ with unknown parameters can still be derived according to the formula (17). According to the chain rule,

$$\frac{\partial W_{cd}^{e}\left(J_{1T},J_{2T}\right)}{\partial\boldsymbol{T}^{\alpha}}=\frac{\partial W_{cd}^{e}}{\partial\left(J_{1T}\right)^{2}}\frac{\partial\left(J_{1T}\right)^{2}}{\partial\boldsymbol{T}^{\alpha}}+\frac{\partial W_{cd}^{e}}{\partial J_{2T}}\frac{\partial J_{2T}}{\partial\boldsymbol{T}^{\alpha}}=\frac{2w_{1t}}{V^{2}}\left(\boldsymbol{N}_{\alpha}\otimes\boldsymbol{N}_{\beta}\right)\cdot\boldsymbol{T}^{\beta}+\frac{2w_{2t}}{V^{2}}p_{\alpha\beta}\boldsymbol{T}^{\beta}\,, \tag{24}$$

where

$$w_{1t}=\partial W_{cd}^{e}/\partial\left(J_{1T}\right)^{2}, \tag{25}$$

$$w_{2t}=\partial W_{cd}^{e}/\partial J_{2T}\,. \tag{26}$$

Similarly,

$$\frac{\partial W_{cd}^{e}\left(J_{1T},J_{2T}\right)}{\partial\theta}=\frac{\partial W_{cd}^{e}}{\partial\left(J_{1T}\right)^{2}}\frac{\partial\left(J_{1T}\right)^{2}}{\partial\theta}=\frac{2w_{1t}}{V^{2}}\left(\boldsymbol{T}^{\alpha}\cdot\boldsymbol{N}_{\alpha}\right)\left(\boldsymbol{T}^{\beta}\cdot\frac{\mathrm{d}\boldsymbol{N}_{\beta}}{\mathrm{d}\theta}\right). \tag{27}$$

In formula (16),

$$\boldsymbol{N}_\alpha \cdot \boldsymbol{N}_\beta = \boldsymbol{P}_\alpha \cdot \boldsymbol{P}_\beta , \tag{28}$$

which makes $\partial J_{2T} / \partial \theta = 0$ in formula (27). The parameters w_{1t} and w_{2t} are functions with J_{1T} and J_{2T} as basic unknowns. According to formula (24), it is not difficult to see that the element compliance matrix derived from formula (17) is

$$\boldsymbol{C}_{\alpha\beta} = \frac{2}{V^2}\left[w_{1t}\left(\boldsymbol{N}_\alpha \otimes \boldsymbol{N}_\beta\right) + w_{2t} p_{\alpha\beta} \boldsymbol{I} \right]. \tag{29}$$

Because the complementary energy function W_{cd}^e can take any form with J_{1T} and J_{2T} as independent variables, formula (29) is a general expression for the element compliance matrix. When selecting different materials, w_{1t} and w_{2t} only need to be changed. If

$$w_{1t} = -\frac{\nu V}{2E}, w_{2t} = \frac{(1+\nu)V}{2E}, \tag{30}$$

$\boldsymbol{C}_{\alpha\beta}$ and $\partial W_{cd}^e / \partial \theta$ are suitable for linear elastic materials.

BP neural network embedded in BFEM model

According to the derivation in Section 2, the parameters w_{1t} and w_{2t} are crucial for describing the constitutive relationship. They are functions with J_{1T} and J_{2T} as basic unknowns:

$$w_{1t} = w_{1t}\left(J_{1T}, J_{2T}\right), \quad w_{2t} = w_{2t}\left(J_{1T}, J_{2T}\right). \tag{31}$$

This mapping is described by a BP neural network, with J_{1T} and J_{2T} as inputs and w_{1t} and w_{2t} as outputs. The expression for a BP neural network with n layers is

$$\begin{aligned} &\boldsymbol{x}_1 = \left[J_{1T}, J_{2T}\right]^{\mathrm{T}} \\ &\boldsymbol{x}_2 = f_2\left(\boldsymbol{H}^{(2,1)} \cdot \boldsymbol{x}_1 + \boldsymbol{b}^{(2,1)}\right) \\ &\mathrm{M} \\ &\boldsymbol{x}_{\mathrm{n}} = f_{\mathrm{n}}\left(\boldsymbol{H}^{(\mathrm{n,n-1})} \cdot \boldsymbol{x}_{n-1} + \boldsymbol{b}^{(\mathrm{n,n-1})}\right) = \left[w_{1t}, w_{2t}\right]^{\mathrm{T}} \end{aligned}, \tag{32}$$

where $\boldsymbol{H}^{(\mathrm{i,i-1})}$ is the weight matrix, $\boldsymbol{b}^{(\mathrm{i,i-1})}$ is the deviation vector, and f_{i} is the activation function. Their superscripts i and i-1 represent the layer number, i = 2,3... n. The activation function for each layer is

$$f_{\mathrm{i}}(y) = \begin{cases} \dfrac{2}{1+e^{-2y}} - 1 & \mathrm{i} < n \\ y & \mathrm{i} = n \end{cases}. \tag{33}$$

They enable the neural network to have nonlinear mapping ability. During the training process of neural networks, the weight matrix and deviation vector of each layer are updated using the gradient descent method until the error function is less than an acceptable error. If there are m sets of reference data, the error function is

$$\mathrm{Err}\left(\boldsymbol{H}, \boldsymbol{b}\right) = \frac{1}{2m}\sum_{k=1}^{m}\left[\left(\hat{w}_{1t}^k - w_{1t}^k\right)^2 + \left(\hat{w}_{2t}^k - w_{2t}^k\right)^2\right], \tag{34}$$

where $\hat{w}_{1t}^k$ and $\hat{w}_{2t}^k$ are the predicted values of the neural network, and w_{1t}^k and w_{2t}^k are the reference values. The training process of the BP neural network and the calculation process of the BFEM model are shown in Fig. 1, and $\boldsymbol{c}$ is a set of weight matrices and deviation vectors for each layer. When iteratively updating the basic unknowns, it is necessary to recalculate J_{1T} and J_{2T} for each element to obtain the predicted values of w_{1t} and w_{2t}. The predicted values are substituted into equation (23) to test whether the new basic unknowns satisfy the control equations.

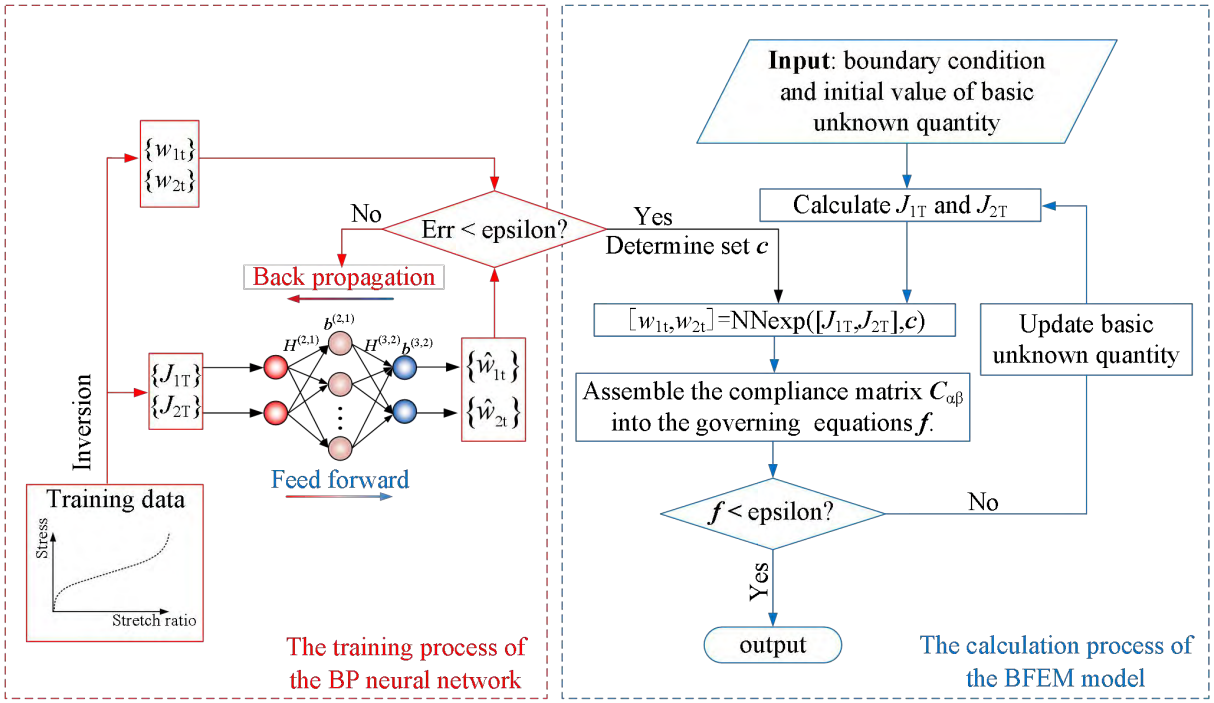


Figure 1. Training of BP Neural Network and BFEM solution flowchart

Example: stretching of a plate with a defect in the center

The top of the hyperelastic material plate with a defect in the center is subjected to a uniformly distributed load P. Due to the symmetry of structure and force, half of the plate is used for numerical research. The size, constraints, and mesh generation of the structure are shown in Fig. 2. To provide data and make comparisons, the constitutive relationship of this structure was established using the Mooney-Rivlin model [19]. This model is represented as

$$W = C_1(I_1 - 3) + C_2(I_2 - 3), \tag{35}$$

where the parameter C_1 is 0.805 MPa and C_2 is 0.194 MPa. The training samples of the neural network were obtained by further inverting the uniaxial stretching data (stretching ratio of 1-6). A pre-trained BP neural network with 2 hidden layers and 7 neurons per layer is used to establish a new constitutive relationship. The mean squared errors of the training set, validation set, and test set are recorded in Fig. 3. It can be seen that the mean squared errors of the three sets are close to 10^{-15}, indicating good training results. Compare the vertical displacement of points A and B (marked in Fig. 2) with the results of the CPS8R element in Abaqus and record them in Fig. 4. The relative error of point A is 4.32%, and the relative error of point B is 3.62% when $P = 2$ MPa.

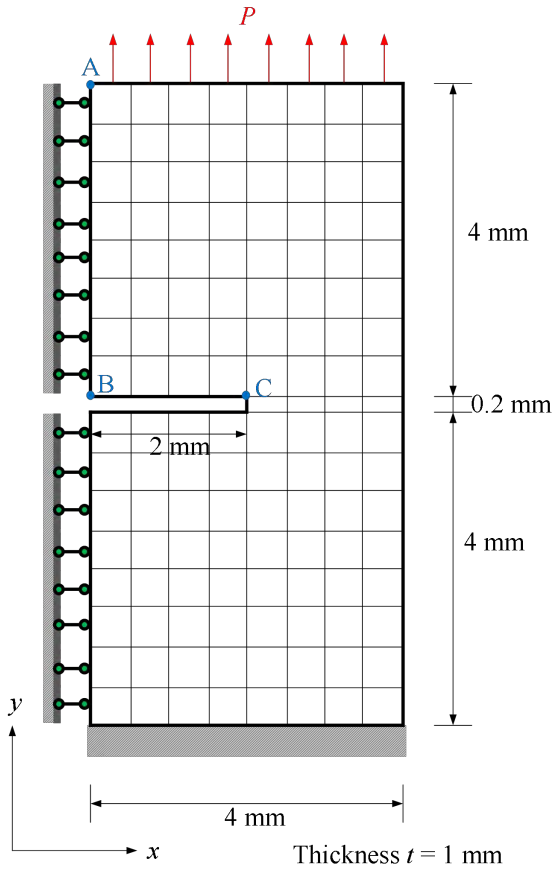


Figure 2. Plate with center defect (1/2 part)

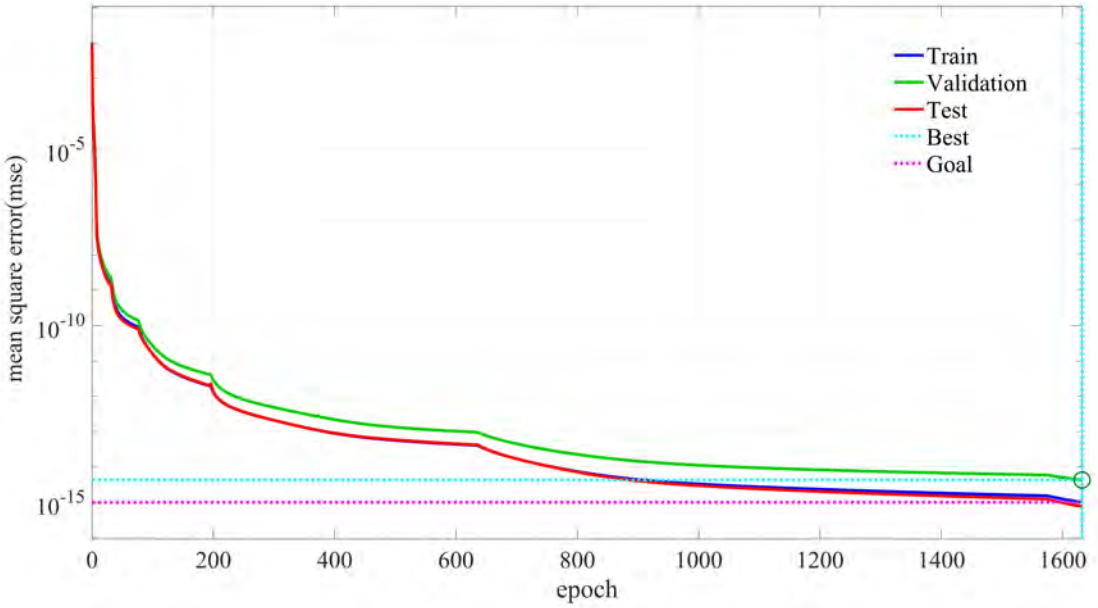


Figure 3. Mean squared error during the BP neural network training process

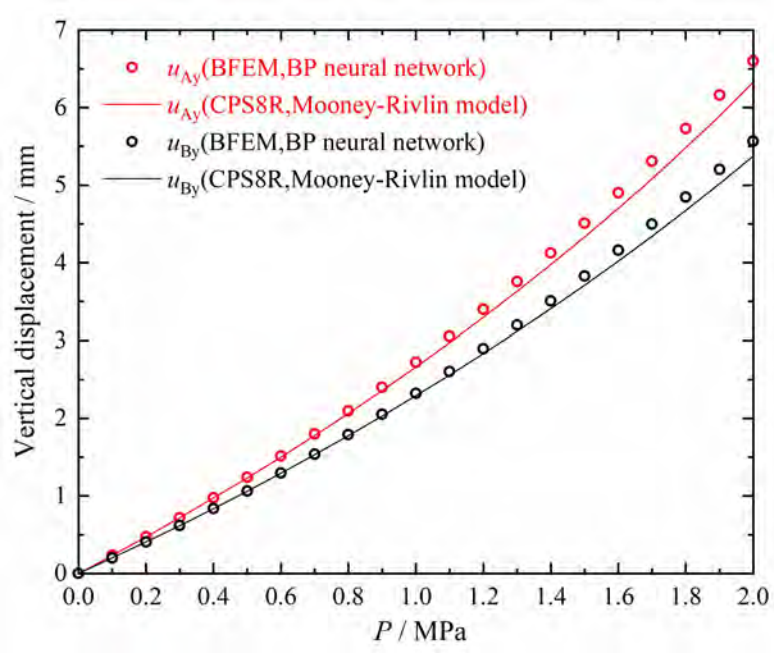


Figure 4. The vertical displacement of points A and B and the results of CPS8R element

As P increases, the stress of each element and the deformation contour are recorded in Fig. 5-7. From these figures, it can be seen that there is a clear phenomenon of stress concentration at the defect. In Fig. 5, there are higher stresses along the positive x-axis at the bottom of the structure and near point C, and higher stresses along the negative x-axis near point B. In Fig. 6, the stress σ_{yy} in the area from point C to point B gradually becomes smaller. In Fig. 7, the direction of stress σ_{xy} is opposite at the upper and lower edges of the defect. The obvious tensile deformation occurs on the overall structure, and the obvious rotation occurs on the elements around the defect (such as the BC edge).

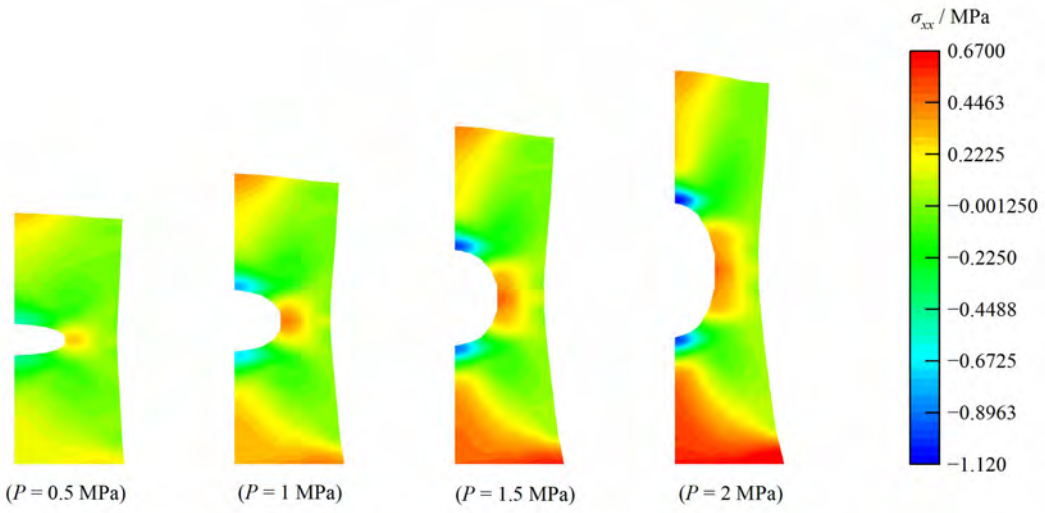


Figure 5. Stress σ_{xx} and deformation contour

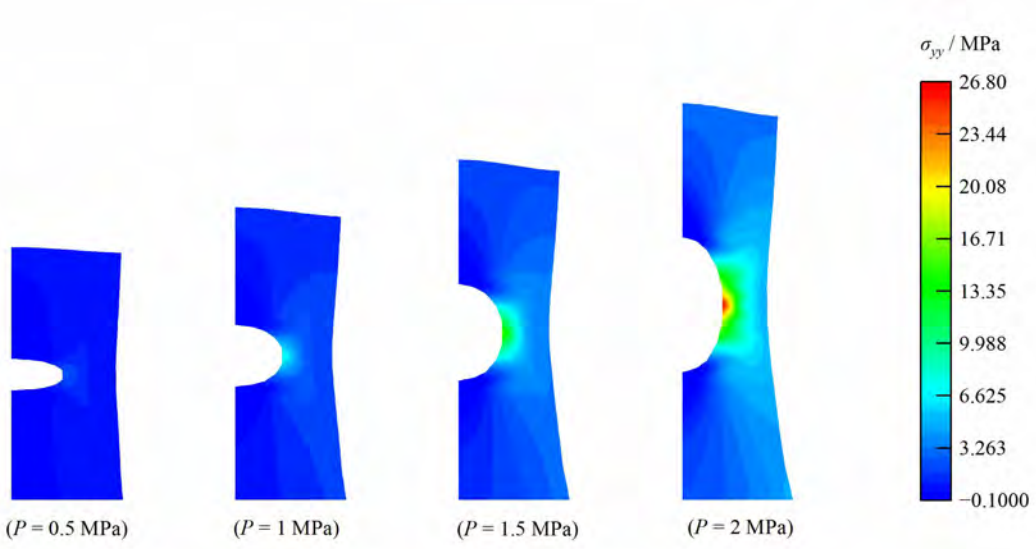


Figure 6. Stress σ_{yy} and deformation contour

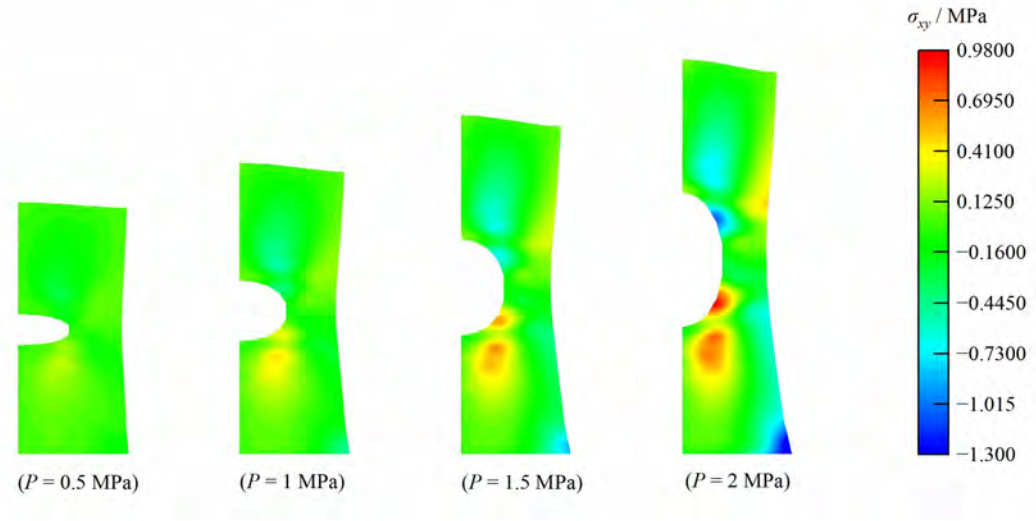


Figure 7. Stress σ_{xy} and deformation contour

Conclusions

A new BFEM based on the complementary energy principle is proposed, which can be used to solve finite strain and finite deformation problems of incompressible hyperelastic materials. The control equations for solid elements are derived by taking stationary values on complementary energy functional. The control equation contains unknown parameters, and the nonlinear mapping between variables and unknown parameters can be replaced by pre-trained BP neural networks. The pre-trained BP neural network model is embedded into the BFEM calculation process to simulate the mechanical response of incompressible hyperelastic materials. The following conclusions were drawn:

1. In the derivation process, formulas are represented by tensors, with concise symbols and easy programming calculations. The stress tensors involved are not approximated by interpolation functions but are expressed using specific formulas composed of base force.
2. Due to the uncertainty of the specific form of the complementary energy function of elements, any form with J_{1T} and J_{2T} as variables can be used. Therefore, the element compliance matrix of hyperelastic materials proposed in this paper is a general expression.

3. The constitutive model of hyperelastic materials is replaced by a pre-trained BP neural network. When BFEM is used to simulate different hyperelastic materials, only the provided samples need to be modified and retrained.
4. The Mooney-Rivlin model is used to provide test samples in numerical calculations. These samples are further inverted to obtain pre-trained BP neural networks. The predicted parameters $\hat{w}_{1t}^{k}$ and $\hat{w}_{2t}^{k}$ are substituted into the general compliance matrix expression for calculation. This enables BFEM to accurately simulate the mechanical response characterized by the Mooney-Rivlin model, which also confirms the strong universality of the compliance matrix expression.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (10972015, 11172015), Beijing Natural Science Foundation (8162008), and the Pre-exploration Project of Key Laboratory of Urban Security and Disaster Engineering, Ministry of Education, Beijing University of Technology (USDE201404).

Reference

[1] E. Reissner, On a Variational Theorem for Finite Elastic Deformations, Studies in Applied Mathematics 32(2) (1953) 129–135.
[2] Y.J. Peng, Y.H. Liu, Advances in the Base Force Element Method, Springer, Singapore, 2019.
[3] Y.C. Gao, A new description of the stress state at a point with applications, Archive of Applied Mechanics 73(3-4) (2003) 171-183.
[4] Y.J. Peng, Z.H. Li, M.M.A. Kamel, Geometrical nonlinear problems of truss beam by base force element method, International Journal for Numerical Methods in Engineering 122(18) (2021) 4793-4824.
[5] Z.H. Li, Y.J. Peng, The base force element method based on the arc-length method for stability analysis, International Journal of Non-Linear Mechanics 144 (2022).
[6] R.S. Rivlin, Large Elastic Deformations of Isotropic Materials. II. Some Uniqueness Theorems for Pure, Homogeneous Deformation, Philosophical Transactions of the Royal Society of London 240(822) (1948) 491-508.
[7] A.N. Gent, A new constitutive relation for rubber, Rubber Chemistry and Technology 69(1) (1996) 59-61.
[8] R.W. Ogden, Non-linear elastic deformations, Engineering Analysis (1984).
[9] M.M. Carroll, A Strain Energy Function for Vulcanized Rubbers, Journal of Elasticity 103(2) (2011) 173-187.
[10] T. Beda, An approach for hyperelastic model-building and parameters estimation a review of constitutive models, European Polymer Journal 50 (2014) 97-108.
[11] T.J. Pence, K. Gou, On compressible versions of the incompressible neo-Hookean material, Mathematics and Mechanics of Solids 20(2) (2015) 157-182.
[12] S.J. Lee, R.T. Shield, VARIATIONAL-PRINCIPLES IN FINITE ELASTOSTATICS, Zeitschrift Fur Angewandte Mathematik Und Physik 31(4) (1980) 437-453.
[13] B. Müller, G. Starke, A. Schwarz, J. Schröder, A FIRST-ORDER SYSTEM LEAST SQUARES METHOD FOR HYPERELASTICITY, Siam Journal on Scientific Computing 36(5) (2014) B795-B816.
[14] T. Kirchdoerfer, M. Ortiz, Data-driven computational mechanics, Computer Methods in Applied Mechanics and Engineering 304 (2016) 81-101.
[15] G. Capuano, J.J. Rimoli, Smart finite elements: A novel machine learning application, Computer Methods in Applied Mechanics and Engineering 345 (2019) 363-381.
[16] L.T.K. Nguyen, M.A. Keip, A data-driven approach to nonlinear elasticity, Computers & Structures 194 (2018) 97-115.
[17] Y.C. Gao, Z. Wang, Complementary energy principle for large elastic deformation, Science in China Series G-Physics Mechanics & Astronomy 49(3) (2006) 341-356.
[18] Y.J. Peng, Y.H. Liu, Base force element method of complementary energy principle for large rotation problems, Acta Mechanica Sinica 25(4) (2009) 507-515.
[19] M.J. Mooney, A Theory of Large Deformation. Journal of Applied Physics, Journal of Applied Physics 11(9) (1940) 582-592.

Further accuracy verification of the 2D adaptive mesh refinement method by the benchmarks of lid-driven cavity flows

Rajnesh Lal[1], *†Zhenquan Li[2] and Miao Li[3]

[1]School of Mathematical and Computing Sciences, Fiji National University, Lautoka, Fiji
[2]School of Computing and Mathematics, Charles Sturt University, Thurgoona, NSW 2640, Australia
[3]CSU Engineering, Charles Sturt University, Bathurst, NSW 2795, Australia
*Presenting author: jali@csu.edu.au
†Corresponding author: jali@csu.edu.au

Abstract

The lid-driven cavity flow problem is a well-known test case in fluid dynamics for validating computational fluid dynamics (CFD) algorithms. Despite its geometrical simplicity, the lid-driven cavity flow problem exhibits a complex flow regime, mainly due to the vortices formed in the centre and at the corners of the square domain. Consequently, this paper verifies the accuracy of a 2D adaptive mesh refinement (AMR) method in estimating the locations of the centre of vortices for a steady incompressible flow in a 2D lid-driven square cavity. We consider an initial coarse uniform grid mesh with a resolution of 20×20 and 50×50 for Reynolds number $Re = 1000$ and 2500, respectively and perform ten refinements. Our study reports the location of the centre of vortices obtained for $Re = 1000$ and 2500. The accuracy of the result is shown by comparing the coordinates of centres of vortices located by the AMR method with the corresponding benchmark results from four different literature.

Keywords: Adaptive mesh refinement, Refined mesh, Lid driven cavity, Centre of vorticies.

Introduction

Computational Fluid Dynamics (CFD) has evolved into a powerful tool for simulating fluid flow, heat transfer, and other related phenomena in diverse engineering applications. The discretization of the computing domain through a mesh is crucial to accurate and robust CFD simulations. Mesh refinement, a crucial aspect of this discretization process, involves adjusting the grid resolution to capture fine details and complex flow features.

Mesh generation and adaptivity remain major obstacles in the CFD workflow. Ensuring consistent and dependable convergence of flow solvers and residual values poses challenges in various industrial scenarios [1]. While numerous CFD software packages can achieve convergence in simpler scenarios, they often struggle with flows involving difficult flow physics and/or complex geometries, such as those encountered in high-lift aircraft configurations. Many existing solver techniques lack the robustness required to guarantee reliable convergence in such cases. Adaptive mesh refinement (AMR) is a computational strategy allowing users to optimise their simulations by dynamically modifying the resolution of the computational mesh in accordance with evolving flow characteristics and phenomena, thereby decreasing computational cost. AMR techniques often adjust the computational mesh during the time-stepping process, refining it selectively in areas of interest or where refinement is deemed necessary [2, 3]. Although AMR strategies offer the potential for increased accuracy at a lower computational cost, their adoption has been limited by concerns regarding robustness, error estimation, management of complex geometries, and software complexity issues [1].

In tackling the complexities of accurately simulating fluid flow problems while maintaining computational efficiency, the AMR technique has undergone significant evolution, driven by contributions from various researchers in the CFD domain. The origin of adaptive mesh refinement for CFD can be traced back to the 1980s and 1990s, encompassing seminal works by researchers such as Berger and Oliger [4], Bell *et al*. [5], Friedel *et al*. [6], and Berger and Leveque [7]. Since then, the development of AMR has continued to evolve with advancements in numerical methods, computer hardware and software tools, and applied to a variety of fluid dynamics and computational science problems.

Li [8] introduced a 2D AMR technique derived from the qualitative theory of differential equations. This AMR method refines a computational mesh using numerically computed velocity fields. The efficacy and accuracy of this AMR method have been verified through the accurate locations of singular points, asymptotic lines, and closed streamlines [9], as well as through comparisons with established CFD benchmark experiments up to two refinements, including lid-driven cavity flow [10], 2D unsteady flow past a square cylinder [11], backwards-facing step flow [12], and 2D flow over a wall-mounted plate [13]. Furthermore, the AMR method has demonstrated its capability to capture localized flow features, such as identifying accurate locations of the centre of vortices within refined cells [10, 13]. The accuracy of the AMR method depends solely on the accuracy of the numerical methods used. The AMR method is a low-cost and robust method applicable to all incompressible fluid flows [10, 2].

The lid-driven cavity flow problem is a well-known test case in fluid dynamics for validating CFD algorithms. Even though the lid-driven cavity flow problem appears geometrically simple, it exhibits a complex flow regime primarily attributable to the formation of vortices at the centre and corners of the square domain. The combination of its simple geometry, the presence of various corner singularities, and the difficulty in approximating the exact solution made the lid-driven cavity one of the most common benchmarks in CFD. Several benchmark results are available in the literature, including the works of Ghia *et al*. [14], Botella and Peyret [15], Erturk *et al*. [16], and Shapeev and Lin [17], among others. Ghia *et al*. [14] used the vorticity-stream function formulation of the 2D incompressible Navier-Stokes (N-S) equations to study the effectiveness of the coupled strongly implicit multigrid method. The solution for flows with several Reynolds numbers (Re) is computed. For $Re = 1000$, the fine-mesh flow solution using the uniform grid mesh of resolution 129×129 is reported. Botella and Peyret [15] reported the solutions for the lid-driven cavity flow for $Re = 1000$, computed by a Chebyshev collocation method with a grid mesh of $N = 160$ (polynomial degree). Erturk *et al*. [16] have used the N-S equations in stream function and vorticity formulation, which are solved numerically using a fine uniform grid mesh of 601×601. The solution for flows computed with several high $Re \leq 21000$ are reported. Shapeev and Lin [17] have used the stream function formulation of the N-S equations for the solution of the lid-driven cavity flow with several Re. In [17], the problem is approached using a high-order finite element technique featuring exponential mesh refinement in proximity to the corners, coupled with analytical asymptotics of the flow near the corners. For $Re = 1000$ and 2500, the results using 49156 triangular elements are presented in [17].

The earlier works, e.g. [2, 10], examined the accuracy of the 2D AMR method with two refinements and used the finite volume methods to solve the N-S equations numerically. As the 2D AMR technique can be iteratively applied, greater accuracy in locating vortex centres can be achieved if numerical velocity fields are sufficiently accurate and additional refinements are implemented. Hence, the present study further verifies the accuracy of a 2D AMR method in estimating the locations of the centre of vortices for a steady incompressible flow in a 2D

lid-driven square cavity using additional mesh refinements. A finite element method is used to solve the N-S equations numerically. We consider an initial coarse uniform grid mesh with a resolution of 20×20 and 50×50 for Reynolds number $Re = 1000$ and 2500, respectively and perform ten refinements. The accuracy of the results obtained for $Re = 1000$ and 2500 are demonstrated by comparing the coordinates of centres of vortices located by the AMR method with the corresponding benchmark results from those presented in [14, 15, 16, 17].

Method

This section describes the governing equations, the computational domain and the mesh structure, the flow solver, and briefly on the 2D AMR method.

The governing equations and computational domain

The present study consider the finite element discretizations of the 2D steady, incompressible Navier-Stokes equations, which are governed by the continuity equation and the momentum equations in two directions defined by:

$$\begin{aligned} \nabla \cdot \mathbf{V} &= 0 \\ -\nu \nabla^2 \mathbf{V} + \mathbf{V} \cdot \nabla \mathbf{V} + \frac{1}{\rho} \nabla P &= 0, \end{aligned}$$

where $\mathbf{V} = (u, v)$ denotes the velocity field in 2D with u as the velocity component at the x-direction and v as the velocity component at the y-direction, ν is the kinematic viscosity, ρ is the fluid density, and P represents the scalar pressure.

The lid-driven cavity consists of a square cavity with the dimensions of L. At the top boundary, a tangential unit velocity ($u = 1, v = 0$) is applied to drive the fluid flow in the cavity. The remaining three rigid walls are imposed with no-slip boundary conditions ($u = v = 0$). The origin of the cartesian coordinate is located at the left lower corner of the cavity. The geometry of the computational domain is shown in Figure 1. The Reynolds number is defined as $Re = UL/\nu$, where U is the velocity of the lid, L is the length of the cavity, and ν is the kinematic viscosity of the fluid.

Computational mesh and the flow solver

For the case with $Re = 1000$, the initial mesh (Mesh_0) is a uniform grid mesh with resolution of 20×20, i.e. Mesh_0 has 441 computational nodes. This corresponds to 1323 degrees of freedom (DOF). The DOF is given by the number of nodes multiplied by the number of dependent variables. For $Re = 2500$, Mesh_0 is a uniform grid mesh with a resolution of 50×50 corresponding to 2601 computational nodes and 7803 DOF. The `Finite Element Analysis simulation Toolbox` (`FEATool`, version 1.16.3, https://www.featool.com/) in `MATLAB` (R2023a) was used to numerically solve the steady 2D Navier-Stokes equations. We assumed that the convergence is achieved when the numerical solutions (velocity fields) are computed with residuals smaller than 10^{-12}.

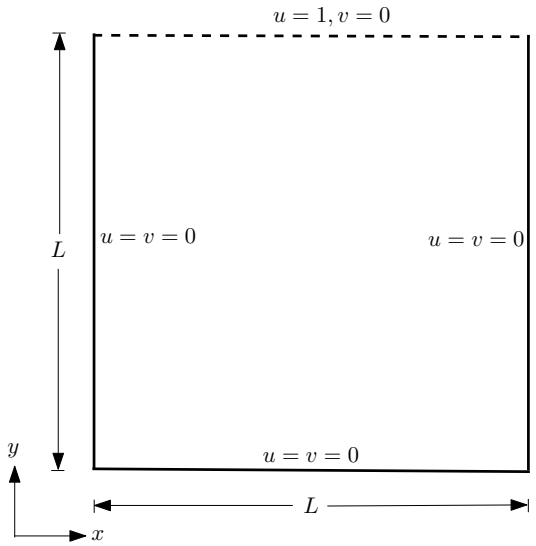


Figure 1: Cavity flow geometry and boundary conditions.

The 2D adaptive mesh refinement method

In the present study, we apply the same AMR method from [13]. We summarise the 2D AMR method and further refer the reader to Section 3 of [13] for a detailed description of the method.

Let $\mathbf{V}_l = \mathbf{AX}+\mathbf{b}'$ be defined as a vector field on a triangle obtained through linear interpolation of vectors at the three vertices of the triangle in the domain of the velocity field, where

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}, \ \mathbf{b}' = \begin{pmatrix} b'_1 \\ b'_2 \end{pmatrix}, \text{ and } \mathbf{X} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

is a matrix of constants, a vector of constants, and a vector of spatial variables, respectively.

For the interpolated vector field $\mathbf{V}_l$ to satisfy the continuity equation for incompressible or steady-state flow, the equation $\nabla \cdot \mathbf{V}_l = \text{trace}(\mathbf{A}) = 0$ must be satisfied. However, the interpolated numerical velocity vector field generally does not satisfy the continuity equation. Hence, it is assumed that $f\mathbf{V}_l$ satisfies the continuity equation, where f is an unknown scalar function of the spatial variables x_1 and x_2. A differential equation is obtained when $f\mathbf{V}_l$ is substituted into the vector field $\mathbf{V}$ of the continuity equation $\nabla \cdot \mathbf{V} = 0$, resulting in $\nabla \cdot f\mathbf{V}_l = 0$. Solving the resulting differential equation for the four different Jacobian forms of the constant matrix $\mathbf{A}$ results in four distinct expressions of the function f. Table 1 shows the summary of the Jacobian forms of the constant matrix $\mathbf{A}$ with their corresponding distinct expressions of the function f for the four distinct cases in which the linear interpolations of the vector fields over triangular domains do not satisfy the law of mass conservation.

In Table 1, $(y_1, y_2)^T = \mathbf{V^{-1}X}$ and $(b_1, b_2)^T = \mathbf{V^{-1}b}$ where $\mathbf{V}$ satisfies $\mathbf{AV} = \mathbf{VJ}$, and $\mathbf{J}$ is one of the Jacobian matrices in the table. For $f \notin \{0, \infty\}$, the vectors $\mathbf{V}_l$ and $f\mathbf{V}_l$ produce same streamlines (for more details we refer the readers to Section 2.2 of [9]). Consequently, mass conservation (MC) conditions are that the computed functions f in Table 1 are not equal to zero or infinity at any point on the triangular domains.

A cell-by-cell AMR approach is performed on the quadrilateral grid meshes. A quadrilateral cell on which one of the conditions MC is not satisfied is subdivided. A cell is refined by connecting the mid-points of opposite sides into four equally smaller quadrilateral cells. Applying the AMR

Table 1: Jacobian forms of the constant matrix A and the corresponding distinct expressions of the function f ($C \neq 0$) for all possible cases of a non-mass conservative linear field. The value of C is set to 1 when the conditions (MC) are implemented in `MATLAB`.

Case	Jacobian	f
1	$\begin{pmatrix} r_1 & 0 \\ 0 & r_2 \end{pmatrix}$ $(0 \neq r_1 \neq r_2 \neq 0)$	$\frac{C}{\left(y_1+\frac{b_1}{r_1}\right)\left(y_2+\frac{b_2}{r_2}\right)}$
2	$\begin{pmatrix} r_1 & 0 \\ 0 & 0 \end{pmatrix}$ $(r_1 \neq 0)$	$\frac{C}{y_1+\frac{b_1}{r_1}}$
3	$\begin{pmatrix} r_1 & 0 \\ 0 & r_1 \end{pmatrix}$ $(r_1 \neq 0)$	$\frac{C}{\left(y_1+\frac{b_1}{r_1}\right)^2}$
4	$\begin{pmatrix} \mu & \lambda \\ -\lambda & \mu \end{pmatrix}$ $(\mu \neq 0, \lambda \neq 0)$	$\frac{C}{\left(y_1+\frac{\mu b_1-\lambda b_2}{\mu^2+\lambda^2}\right)^2+\left(y_2+\frac{\lambda b_1+\mu b_2}{\mu^2+\lambda^2}\right)^2}$

method to the initial mesh, Mesh_0, produces the first refined mesh, Mesh_1, and by repeating the procedure nine more times, we obtain the tenth refined mesh, Mesh_{10}. The coordinates of the centre of the isolated refined cells in the area of interest are then taken as an estimate of the centre of vortices.

Numerical results and accuracy verification

This section shows the tenth refined meshes, Mesh_{10}, and the vortices' locations generated in a driven cavity flow for $Re = 1000$ and 2500. The abbreviations BR, BL and TL refer to the bottom right, bottom left and top left, respectively. The number following the abbreviations refers to the vortices that appear in the flow, numbered in order of decreasing size. The abbreviation PV refers to the primary vortex. We then compare estimated centres of PV, BR1, BL1, BR2, BL2 and TL1 with the benchmark results.

The refined meshes, Mesh$_{10}$

Figure 2 shows the tenth refined mesh, Mesh_{10}, for the flow with $Re = 1000$. Figure 3(a)-(e) shows the isolated refined cells containing the centres of vortices in the zoomed-in sections of PV, BR and BL. In Figure 3, red circles are drawn for illustration, and they contain the location of the centres of PV, BR1, BL1, BR2 and BL2 for $Re = 1000$. To validate the local accuracy of the numerically computed velocity fields for the case with $Re = 1000$, in Figures 4 and 5, we present the u-velocity profile along a vertical line and the v-velocity profile along a horizontal line passing through the geometric centre of the cavity, respectively. These profiles agree well with Erturk *et al.* [16], as shown by the symbols in Figures 4 and 5.

For the flow with $Re = 2500$, a similar mesh refinement pattern as for the flow with $Re = 1000$ was observed. Figure 6 shows Mesh_{10} for the flow with $Re = 2500$. The refined mesh in Figure 6 identifies the location of an additional secondary vortex, TL1, which appears near the top left corner. Figures 7 and 8 compare the u and v-velocity profiles, respectively, with the benchmark results of Erturk *et al.* [16]. We observe a good agreement between the numerically computed velocity fields and those reported in [16].

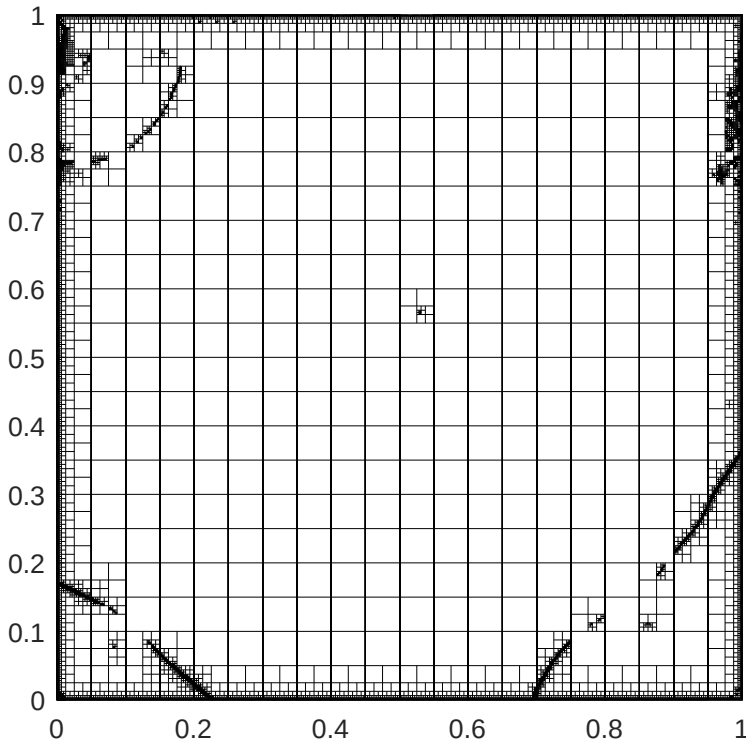


Figure 2: The tenth refined mesh, Mesh$_{10}$, for the flow with $Re = 1000$.

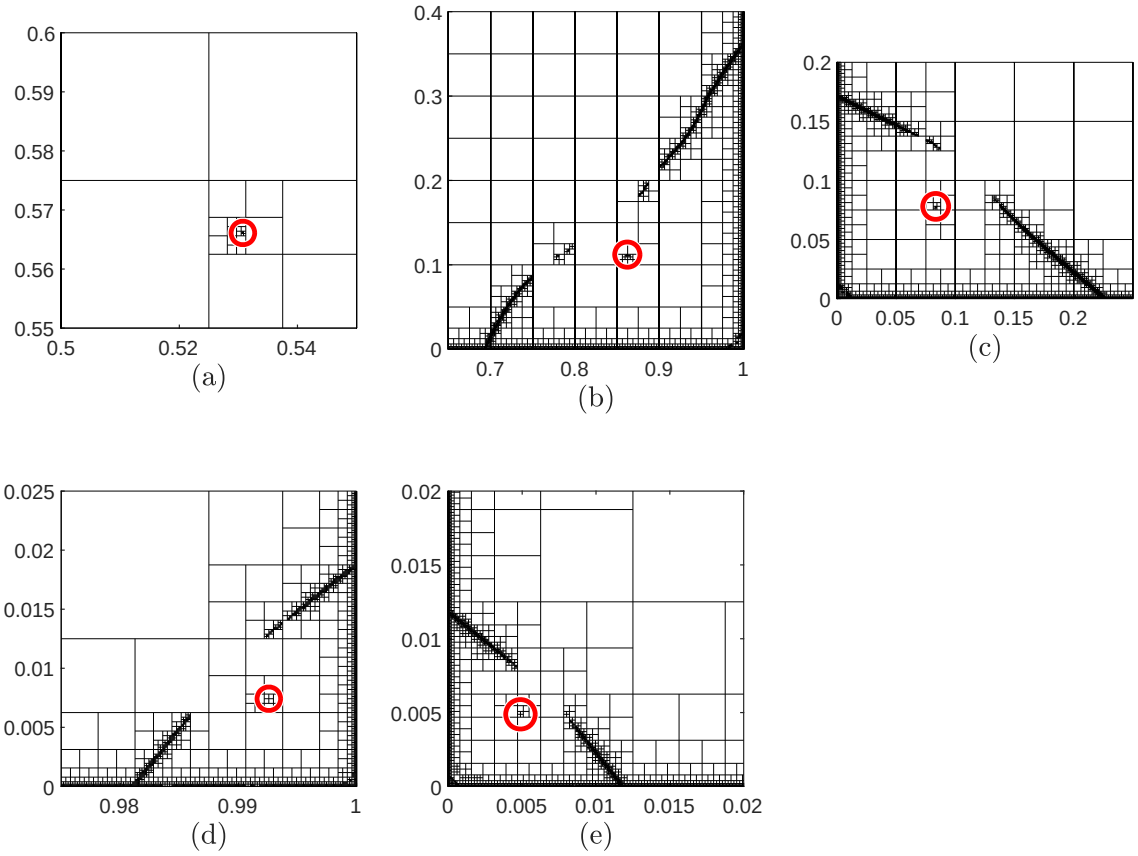


Figure 3: Zoomed-in sections of PV, BR and BL for $Re = 1000$. Red circles contain the location of the centres of (a) PV, (b) BR1, (c) BL1, (d) BR2 and (e) BL2.

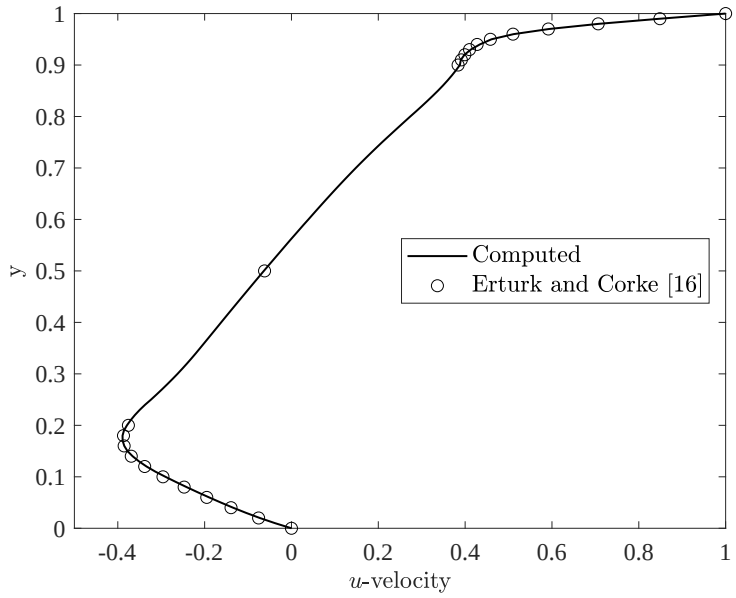


Figure 4: Computed u-velocity profile along a vertical line passing through the geometric centre of the cavity for the flow with $Re = 1000$.

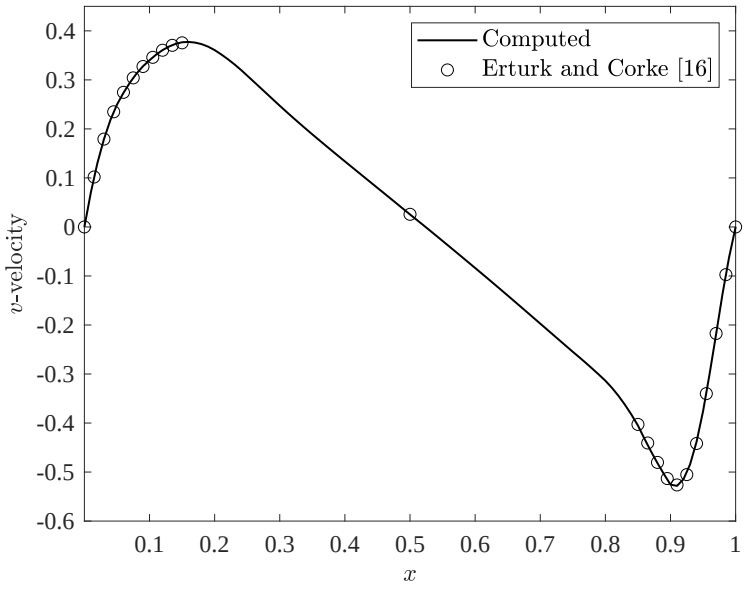


Figure 5: Computed v-velocity profile along a horizontal line passing through the geometric centre of the cavity for the flow with $Re = 1000$.

Vortex center locations

Table 2 presents the estimated coordinates of vortex centres for the flow with $Re = 1000$. The table compares the estimated coordinates with the four reference centre locations reported in [14, 15, 16, 17]. Also, the parameters, such as the grid mesh size and DOF from the corresponding studies, are shown in Table 2. The results obtained by the present method are in good agreement with the four reference centre locations. The best agreement of our results for $Re = 1000$ is with the results of Botella and Peyret [15], Erturk *et al*. [16], and Shapeev and Lin [17]. The relative errors in the centre location of PV, BR1, BL1 and BR2 between our work and [15, 16, 17] is less than 0.0024, and the relative error in the centre location of BL2 is less than 0.03.

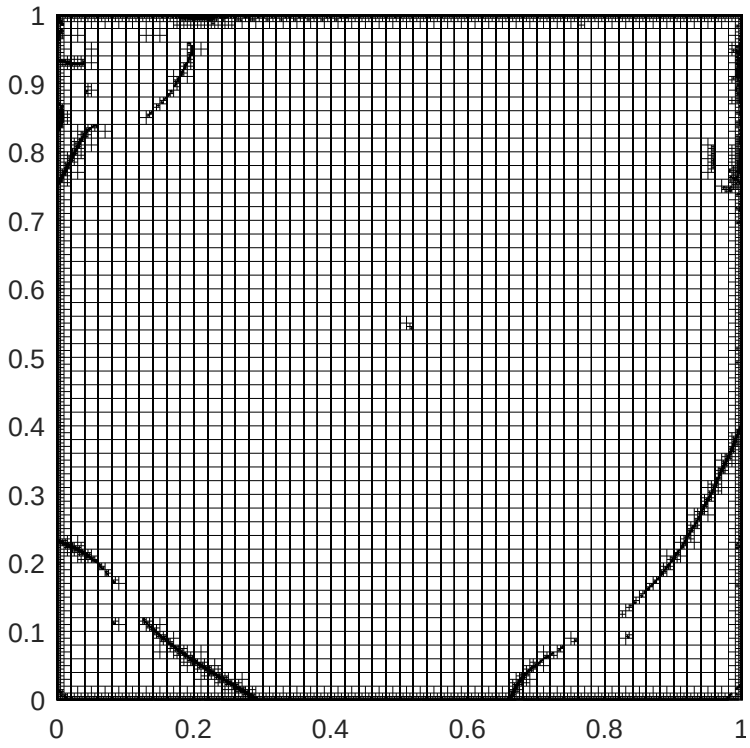


Figure 6: The tenth refined mesh, Mesh$_{10}$, for the flow with $Re = 2500$.

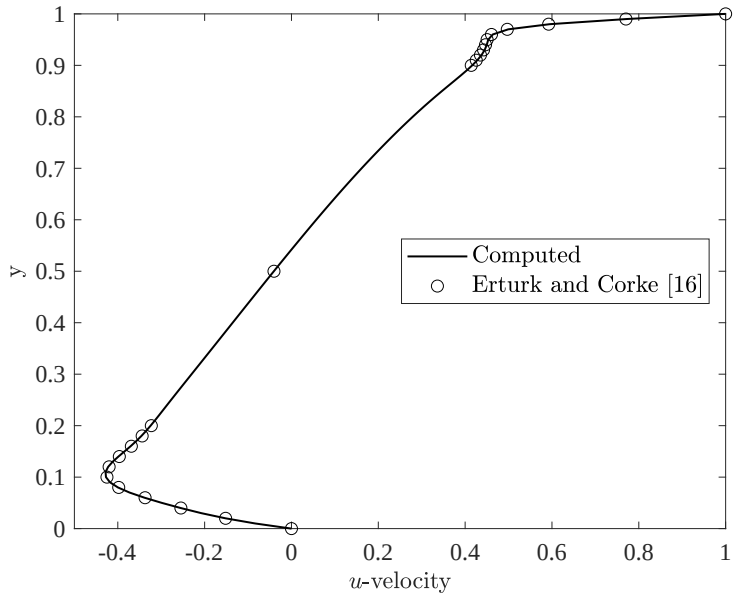


Figure 7: Computed u-velocity profile along a vertical line passing through the geometric centre of the cavity for the flow with $Re = 2500$.

Table 3 tabulates the estimated coordinates of vortex centres for the flow with $Re = 2500$. It compares it with the two reference centre locations reported in Erturk *et al*. [16], and Shapeev and Lin [17]. The centre locations obtained by the present method are in good agreement with that of Erturk *et al*. [16], and Shapeev and Lin [17]. The best agreement of our results for $Re = 2500$ is with the results of Shapeev and Lin [17], where the relative errors in the centre location of vortices between our work and [17] is less than 0.006. The relative errors in the centre location of PV, BR1, BL1, BR2 and TL1 between our work and Erturk *et al*. [16] is less

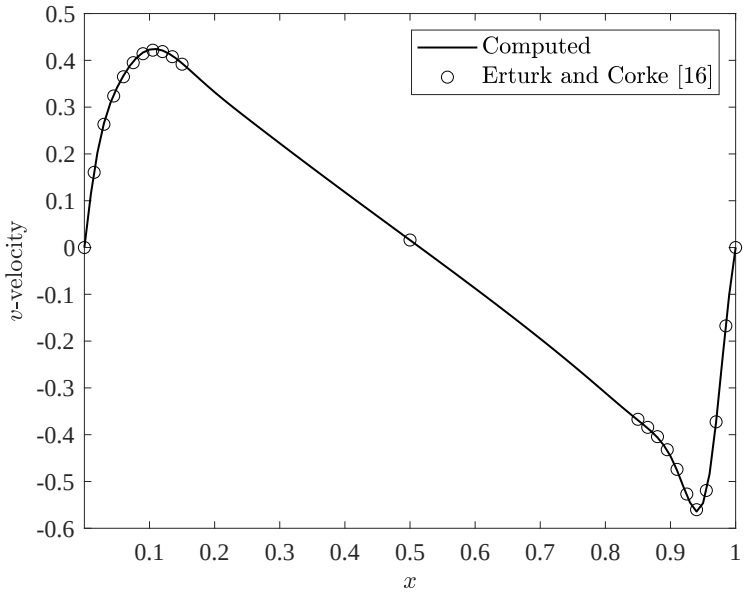


Figure 8: Computed v-velocity profile along a horizontal line passing through the geometric centre of the cavity for the flow with $Re = 2500$.

than 0.

Figure
flows v
the loc
AMR

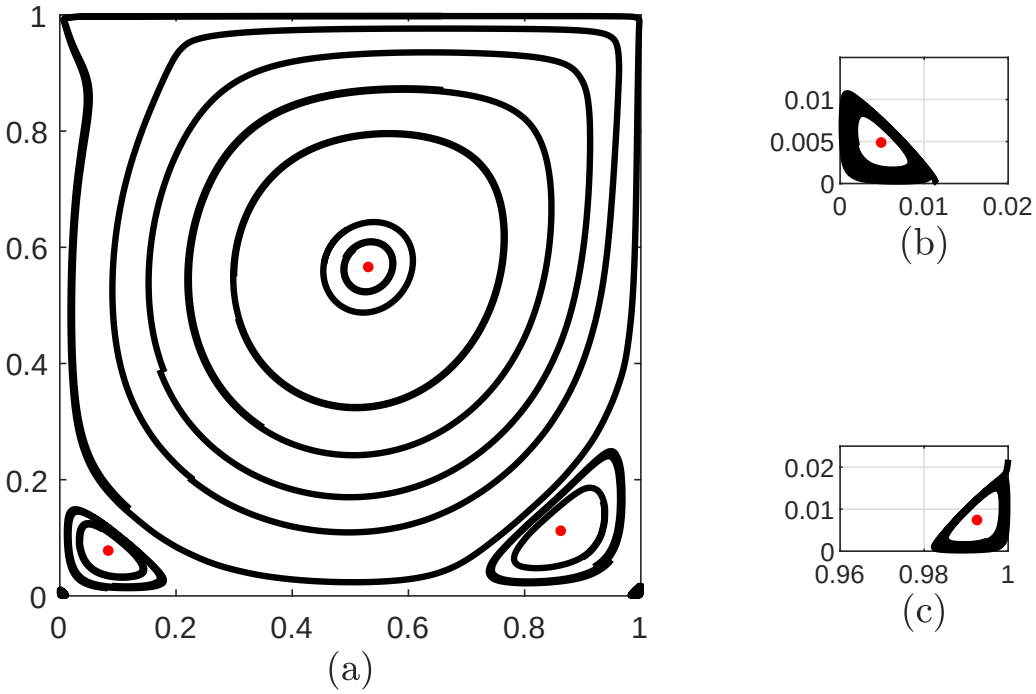


Figure 9: (a) $Re = 1000$: streamline contours and the location of centres of PV, BR1, BL1, BR2, and BL2. The zoomed-in sections of BR and BL show (b) BL2 and (c) BR2, respectively.

Table 2: Location of the centre of vortices for the flow with $Re = 1000$.

Vortex	Reference	Grid size, DOF	Centre location (x, y)
PV	Present	-, 615669	(0.530762, 0.566113)
	Ghia *et al*. [14]	129×129, 33282	(0.5313, 0.5625)
	Botella and Peyret [15]	160×160, 76164	(0.5308, 0.5652)
	Erturk *et al*. [16]	601×601, 722402	(0.5300, 0.5650)
	Shapeev and Lin [17]	-, 226574	(0.530790112, 0.565240557)
BR1	Present	-, 615669	(0.862109, 0.112109)
	Ghia *et al*. [14]	129×129, 33282	(0.8594, 0.1094)
	Botella and Peyret [15]	160×160, 76164	(0.8640, 0.1118)
	Erturk *et al*. [16]	601×601, 722402	(0.8633, 0.1117)
	Shapeev and Lin [17]	-, 226574	(0.86404006, 0.11180617)
BL1	Present	-, 615669	(0.0833008, 0.0780273)
	Ghia *et al*. [14]	129×129, 33282	(0.0859, 0.0781)
	Botella and Peyret [15]	160×160, 76164	(0.0833, 0.0781)
	Erturk *et al*. [16]	601×601, 722402	(0.0833, 0.0783)
	Shapeev and Lin [17]	-, 226574	(0.08327318, 0.078095725)
BR2	Present	-, 615669	(0.992578, 0.00742187)
	Ghia *et al*. [14]	129×129, 33282	(0.9922, 0.0078)
	Botella and Peyret [15]	160×160, 76164	(0.99232, 0.00765)
	Erturk *et al*. [16]	601×601, 722402	(0.9917, 0.0067)
	Shapeev and Lin [17]	-, 226574	(0.992324852, 0.007650979)
BL2	Present	-, 615669	(0.00488281, 0.00488281)
	Ghia *et al*. [14]	129×129, 33282	-
	Botella and Peyret [15]	160×160, 76164	(0.00490, 0.00482)
	Erturk *et al*. [16]	601×601, 722402	(0.0050, 0.0050)
	Shapeev and Lin [17]	-, 226574	(0.0048426963, 0.0048452406)

Conclusions

We further verified the accuracy of the 2D AMR method using two cases of 2D lid-driven cavity flow. A very coarse initial mesh was iteratively refined ten times, and a good estimate of the coordinates of the centre of vortices was obtained for flows considering $Re = 1000$ and $Re = 2500$. The accuracy of the estimates is further supported by the numerically computed u and v velocity profiles, which were in excellent agreement with the benchmark results. Further work will consider applying the 2D AMR method for flows with higher Reynolds numbers.

Table 3: Location of the centre of vortices for the flow with $Re = 2500$

Vortex	Reference	Grid size, DOF	Centre location (x, y)
PV	Present	-,1660092	(0.519629, 0.544043)
	Erturk *et al*. [16]	601×601, 722402	(0.5200, 0.5433)
	Shapeev and Lin [17]	-, 226574	(0.5197769, 0.5439244)
BR1	Present	-, 1660092)	(0.833438, 0.090625)
	Erturk *et al*. [16]	601×601, 722402	(0.8333 , 0.0900)
	Shapeev and Lin [17]	-, 226574	(0.8344014, 0.09075692)
BL1	Present	-, 1660092	(0.0841992, 0.110918)
	Erturk *et al*. [16]	601×601, 722402	(0.0833, 0.1117)
	Shapeev and Lin [17]	-, 226574	(0.08424181, 0.1110061)
BR2	Present	-, 1660092	(0.990469, 0.00935125)
	Erturk *et al*. [16]	601×601, 722402	(0.9900, 0.0083)
	Shapeev and Lin [17]	-, 226574	(0.9904594, 0.009384439)
BL2	Present	-, 1660092	(0.00613281, 0.00621094)
	Erturk *et al*. [16]	601×601, 722402	(0.0067, 0.0067)
	Shapeev and Lin [17]	-, 226574	(0.006129716, 0.006158831)
TL1	Present	-, 1660092	(0.0434375, 0.889062)
	Erturk *et al*. [16]	601×601, 722402	(0.0433, 0.8900)
	Shapeev and Lin [17]	-, 226574	(0.04300225, 0.8893601)

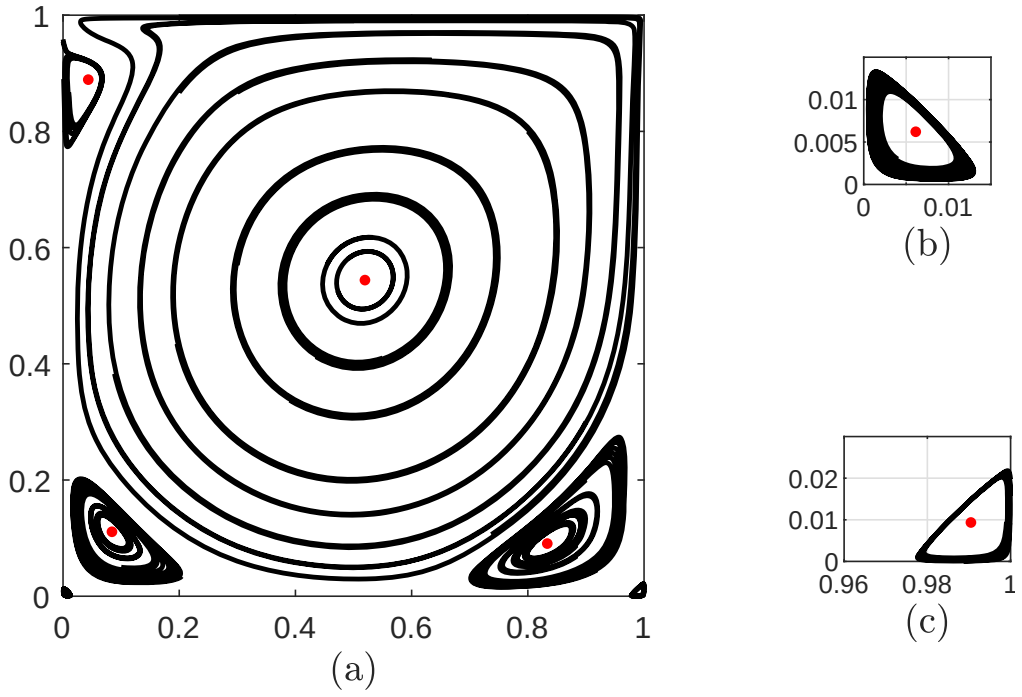


Figure 10: (a) $Re = 2500$: streamline contours and the location of centres of PV, BR1, BL1, BR2, BL2 and TL1. The zoomed-in sections of BR and BL show (b) BL2 and (c) BR2, respectively.

References

[1] Slotnick, J. P., Khodadoust, A., Alonso, J., Darmofal, D., Gropp, W., Lurie, E., and Mavriplis, D. J. (2013) CFD vision 2030 study: a path to revolutionary computational aerosciences. *NASA: Washington, DC, USA*

[2] Li, Z. (2017a) Computational complexity of the algorithm for a 2D adaptive mesh refinement method using lid-driven cavity flows. *Computational Thermal Sciences: An International Journal* **9**, 395–403.

[3] Plewa, T., Linde, T. J., and Weirs, V. G. (2005) Adaptive mesh refinement-theory and applications. In: Proceedings of the Chicago Workshop on Adaptive Mesh Refinement Methods. Springer-Verlag Berlin Heidelberg, 452–456

[4] Berger, M. and Oliger, J. (1984) Adaptive mesh refinement for hyperbolic partial differential equations. *Journal of Computational Physics* **53**, 484–512.

[5] Bell, J., Berger, M., Saltzman, J., and Welcome, M. (1994) Three-dimensional adaptive mesh refinement for hyperbolic conservation laws. *SIAM Journal on Scientific Computing* **15**, 127–138.

[6] Friedel, H., Grauer, R. and Marliani, C. (1997) Adaptive mesh refinement for singular current sheets in incompressible magnetohydrodynamic flows. *Journal of Computational Physics* **134**, 190–198.

[7] Berger, M. J, and Leveque, R. J. (1998) Adaptive mesh refinement using wave-propagation algorithms for hyperbolic systems. *SIAM Journal on Numerical Analysis* **35**, 2298–2316.

[8] Li, Z. (2008) An adaptive two-dimensional mesh refinement method based on the law of mass conservation. *Journal of Flow Visualization and Image Processing* **15**, 17–33.

[9] Li, Z. (2006) An adaptive streamline tracking method for three-dimensional CFD velocity fields based on the law of mass conservation. *Journal of Flow Visualization and Image Processing* **13**, 359–376.

[10] Lal, R. and Li, Z. (2015) Sensitivity analysis of a mesh refinement method using the numerical solutions of 2-D steady incompressible driven cavity flow. *Journal of Mathematical Chemistry* **53**, 844–867.

[11] Li, Z. (2017b) Analysis of 2D unsteady flow past a square cylinder at low Reynolds numbers with CFD and a mesh refinement method. *WSEAS Transactions on Fluid Mechanics* **12**, 150–157.

[12] Li, Z. and Li, M. (2021) Accuracy verification of a 2D adaptive mesh refinement method using backward-facing step flow of low Reynolds numbers. *International Journal of Computational Methods* **18**, 2041012.

[13] Li, Z., and Lal, R. (2023) Application of 2D adaptive mesh refinement method to estimation of the center of vortices for flow over a wall-mounted plate. *International Journal of Computational Methods* **20**, 2143012.

[14] Ghia, U. K. N. G., Ghia, K. N., and Shin, C. T. (1982) High-Re solutions for incompressible flow using the Navier-Stokes equations and a multigrid method. *Journal of Computational Physics* **48**, 387–411.

[15] Botella, O., & Peyret, R. (1998) Benchmark spectral results on the lid-driven cavity flow. *Computers and Fluids* **27**, 421–433.

[16] Erturk, E., Corke, T. C., & Gökçöl, C. (2005) Numerical solutions of 2-D steady incompressible driven cavity flow at high Reynolds numbers. *International Journal for Numerical Methods in fluids* **48**, 747–774.

[17] Shapeev, A. V., and Lin, P. (2009) An asymptotic fitting finite element method with exponential mesh refinement for accurate computation of corner eddies in viscous flows. *SIAM Journal on Scientific Computing* **31**, 1874–1900.

Study on the identification of physical cross-linkages in polymer chains' network

***†I. Riku and K. Mimura**

Department of Mechanical Engineering, Osaka Metropolitan University, Japan

*†riku@omu.ac.jp

Abstract

It is well known that the toughness strength of soft materials such as rubber and elastomer can be improved by addition of chemical cross-linkages or fillers, which will lead to a construction of entangled structure of polymer chains. To enhance such benefit of those added substances, the fraction of them in the entangled structure is expected to be clarified quantitatively. Therefore, in this paper, we at first construct a polymer chains' network with molecular dynamics method. Then, the correlation between the average distance of polymer chains and the average moving velocity of grains is empolyed to identify the existance of physical cross-linkages. Finally, the variation of the identified physical cross-linkages is invesitagated during the deformation of polymer chains' network.

Keywords: Molecular dynamics method, Polymer chains' network, Physical cross-linkage.

Introduction

Soft materials such as rubber and elastomer are mostly used in the cross-linked state of polymer chains. To improve the toughness strength of soft materials, it is usually to add lots of chemical cross-linkages or fillers and as a result, the entangled structures of polymer chains are constructed completely. On the other hand, there are two types of entangled points in the entangled structure of polymer chains: one is the chemical type which is due to the existance of chemical cross-linkages or fillers and obeys the affine deformation; the other one is the physical type, i.e. physical cross-linkages, which is due to the physical interaction between polymer chains and obeys the nonaffine deformation.

To enhance the benefit of the addition of chemical cross-linkages or fillers, the fraction of them in the entangled structure is expected to be clarified quantitatively. However, the count of physical cross-linkages in the entangled structure is not so much easy as the count of chemical cross-linkages or fillers and as a result, the identification of physical cross-linkages has been a pressing issue by now. Therefore, in this paper, we at first construct a polymer chains' network with molecular dynamics method. Then, the correlation between the average distance of polymer chains and the average moving velocity of grains is empolyed to identify the existance of physical cross-linkages. Finally, the variation of the identified physical cross-linkages is invesitagated during the deformation of polymer chains' network.

Molecular Dynamics Model

When one is studying the mechanical properties of a polymeric material, the behavior is mainly governed by the topological features of the entangled structure of polymer chains and using a coarse graining procedure [1] can therefore be a successful approach. In this study, the configuration of polymer chains is represented by a set of the Cartesian coordinates of grains and each grain represents a group of atoms. The bonding potential, U^{bond}, which represents the energy associated with the chemically connected grains separated by a distance r is given by[1]

$$U^{\mathrm{bond}}(r) = U^{\mathrm{FENE}}(r) + U^{\mathrm{LJ}}(r) \tag{1}$$

$$U^{\mathrm{FENE}}(r) = \begin{cases} -\frac{1}{2}kR_0^2 \ln\left[1 - \left(\frac{r}{R_0}\right)^2\right] & r \leq R_0 \\ \infty & r > R_0 \end{cases} \tag{2}$$

$$U^{\mathrm{LJ}}(r) = \begin{cases} 4\epsilon\left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6}\right] & r \leq 2^{1/6}\sigma \\ 0 & r > 2^{1/6}\sigma \end{cases} \tag{3}$$

where k is spring constant, R_0 is finite extended length, σ is Lennard-Jones diameter, ϵ is Lennard-Jones energy. On the other hand, the non-bonding potential, $U^{nonbond}$, which represents the interaction between the grains not chemically connected, or the grains separated far along the polymer backbone is given by

$$U^{\mathrm{nonbond}}(r) = \begin{cases} 4\epsilon\left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6}\right] & r \leq r_c \\ 0 & r > r_c \end{cases} \tag{4}$$

where r_c is the cutoff distance.

Molecular Dynamics Algorithm

The molecular dynamics simulations consider a canonical ensemble (N, V, T), in which N is the total number of grains in the unit cell, V is the volume of the unit cell, T is the temperature. The (N, V, T) ensemble necessitates a modification of the equations of motion with an additional degree of freedom representing a kinetic mass[2][3]

$$\frac{d^2\mathbf{r}_i}{dt^2} = -\frac{\partial U}{\partial \mathbf{r}_i} - \Gamma\frac{d\mathbf{r}_i}{dt} + \mathbf{W}_i(t) \tag{5}$$

where Γ is the friction constant, $\mathbf{W}_i(t)$ is a Gaussian white noise which is generated according to[1]

$$\langle \mathbf{W}_i(t) \cdot \mathbf{W}_j(t') \rangle = \delta_{ij}\delta(t - t')6k_BT\Gamma \tag{6}$$

where k_B is Boltzmann's constant.

Parameters

The molecular dynamics simulations are carried out in the reduced units, in which σ, ϵ and m are taken as the unit of lenght, energy and mass, respectively. The unit of time is given by $\tau = \sigma(m/\epsilon)^{1/2}$. The unit of temperature is given by $T = \epsilon/k_B$. In Table 1, the value of the parameters employed in the simulation is given with the corresponding reduced unit.

Table 1: Value of parameters

σ (σ)	ϵ (ϵ)	k (ϵ/σ^2)	R_0 (σ)	r_c (σ)	Γ (τ^{-1})	m (m)	T (T)
1	1	30	1.5	$2^{1/6}$	0.5	1	1

Boundary Conditions

Fig. 1 shows the configuration of polymer chains in a unit cell after the relaxation computation. The density of the grains consisted in the unit cell is specified as $\rho = 0.85(m/\sigma^3)$, from which the volume of the unit cell can be calculated based on the relation of $V = Nm/\rho$. In this study, the number of grains in each polymer chain is 200, the total number of polymer chains is 50.

To identify the existance of physical cross-linkages in the polymer chains' network shown in Fig. 1, the unit cell is divided by $10 \times 10 \times 10$ elements. In each element, the *average distance* of polymer chains and the *average moving velocity* of grains is to be calculated.

In order to perform the simulation under a cyclic loading condition, the unit cell is elongated and then compressed in the z direction with a constant stretch rate $d\lambda/dt = 10^{-4}\tau^{-1}$. When the unit cell is elongated or compressed in the z direction, the lengths of the unit cell in the x and y directions are changed simultaneously to keep the density of the grains constant. The time step is specified as $dt = 0.01\tau$. Periodic boundary condition is applied to the unit cell throughout the simulation.

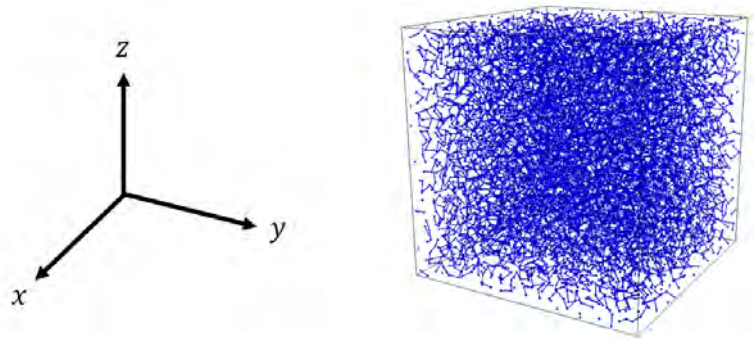


Figure 1: Configuration of polymer chains in the unit cell

Results

Fig. 2 shows the statistical results of the *average distance* of polymer chains in the elements. The number of elements in which the *average distance* of polymer chains is nearer than 1.562 and the number of elements in which the *average distance* of polymer chains is farther than 1.845, are relatively fewer than the others. Therefore, these statistical results can be considered as a normal dispersion.

Fig. 3 shows the statistical results of the *average moving velocity* of grains in the elements. The number of elements in which the *average moving velocity* of grains is slower than 1.380 and the number of elements in which the *average moving velocity* of grains is faster than 1.884, are relatively fewer than the others. Therefore, these statistical results can be considered as a normal dispersion also.

From Fig. 2 and Fig. 3, the correlation between the *average distance* of polymer chains and the *average moving velocity* of grains can be found. Based on this correlation, two constrasting groups of elements, as shown in Table 2, are extracted to identify the existance of physical cross-linkages.

Fig. 4 and Fig. 5 show the configuration of polymer chains inside and outside of one element of group A and group B separately. The polymer chains shown in Fig. 4 distribute densely whereas the polymer chains shown in Fig. 5 distribute nondensely. In other words, the possibility of the existance of physical cross-linkages is quite high in the elements of group A.

Conclusions

In this study, the identification of the existance of physical cross-linkages is tried on based on the correlation between the *average distance* of polymer chains and the *average moving velocity* of grains. A high possibility of the existance of physical cross-linkages is shown in the elements of group A.

References

[1] Kremer, K. and Grest. G. S. (1990) Dynamics of entangled linear polymer melts: A molecular - dynamics simulation. *The Journal of Chemical Physics* **92**, 5057—5086.

[2] Nose, S. (1984) A molecular dynamics method for simulations in the canonical ensemble. *Molecular Physics* **52**, 255—268.

[3] Hoover, W. G. (1985) Canonical dynamics: Equilibrium phase-space distributions. *Physical Review A* **31**, 1695—1697.

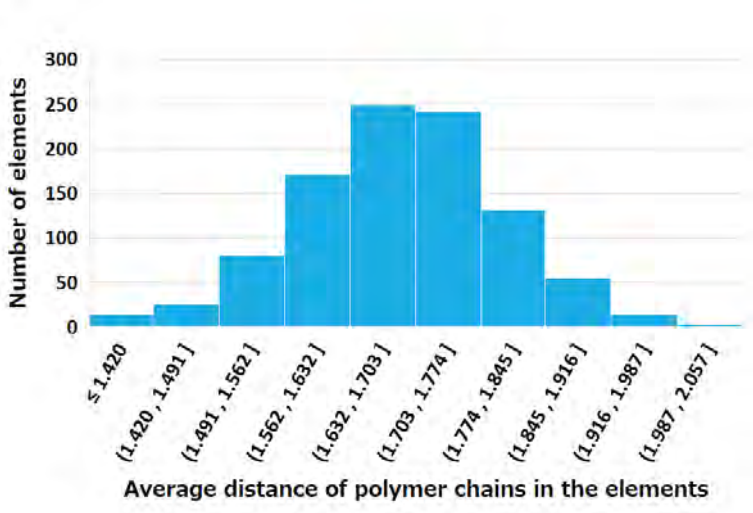


Figure 2: Statistical results of average distance of polymer chains in the elements

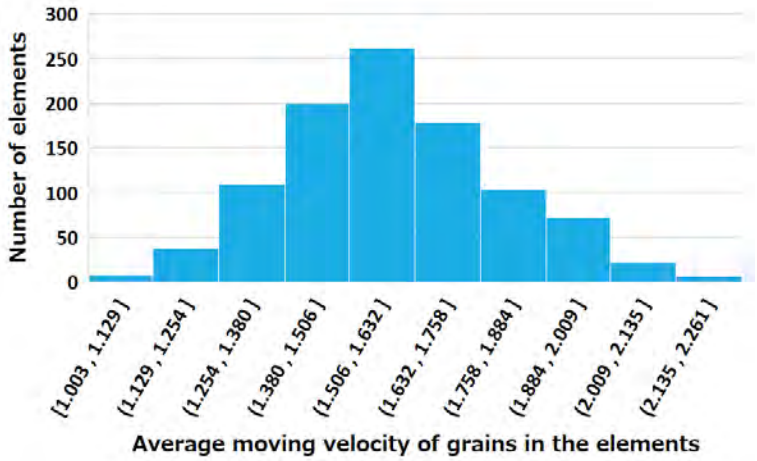


Figure 3: Statistical results of average moving velocity of grains in the elements

Table 2: Extraction of group A and B

	Group A	Group B
Distance	< 1.562	> 1.845
Velocity	< 1.380	> 1.884

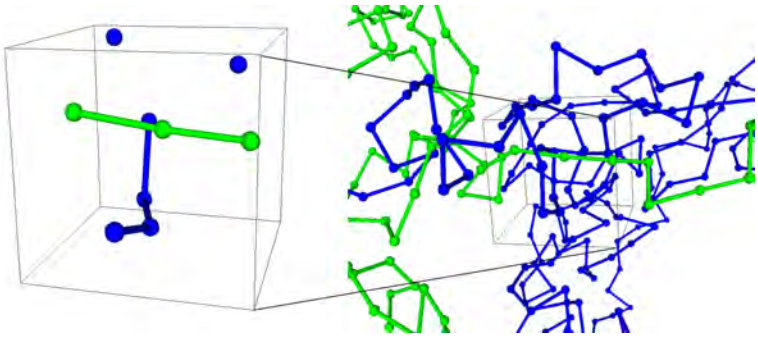


Figure 4: Configuration of polymer chains inside and outside of one element of group A

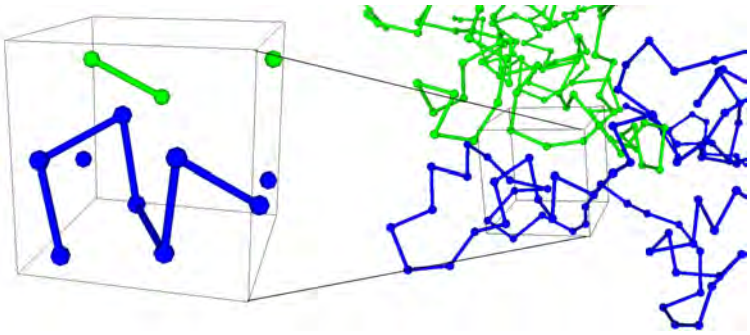


Figure 5: Configuration of polymer chains inside and outside of one element of group B

Identification of effective cohesive parameters for FRP-concrete interface using multi-island genetic algorithm

***Tianxiang Shi[1], Yangyang Wang[1], Miao Pang[1], †Yongqiang Zhang[1]**

[1] College of Civil Engineering and Architecture, Zhejiang University, Hangzhou 310058, China

*Presenting author: 22012256@zju.edu.cn
†Corresponding author: cyqzhang@zju.edu.cn

Abstract

The cohesive zone models (CZMs) are useful tools for describing interface damage and covers a wide range of engineering applications. The formulations of CZMs are expressed by specific constitutive equations, where the related parameters significantly affect the accuracy of the models. To ensure the accurate applications of these models, it is crucial to calibrate the interface parameters, as some of them cannot be directly measured. The identifiability of parameters is verified through an inverse identification framework. The identification problem considers both the experimental and numerical data, including the load-displacement curves and crack length. Due the fact that the debonding in experiments may occur at random locations and expand in various directions, the objective function is modified to identify the complex debonding length. The identification problem is formulated to determine suitable parameters for the fibre reinforced polymer (FRP)-concrete interface in the four-point bending test. The identification results are presented and discussed, aiming to assess the effectiveness of the proposed approach.

Keywords: Cohesive zone model, interfacial debonding, multi-island genetic algorithm

1. Introduction

Nowadays, composite materials have been widely applied in many engineering fields, such as aerospace, biomedical devices, sports, packaging, building and construction [1,2]. With the development of multiphase composites possessing unique properties, studying the properties of these materials becomes critical for the design of new structures.

The interface properties significantly influence the mechanical characteristic of composite materials, as they often initiate damage and facilitate load transfer [3,4]. The debonding at the interface is not visually inspectable in most cases. Thus, understanding the interface failure mechanism in composite materials is very crucial for preventing structures failures. There exist some methods for describing this mechanism, such as virtual crack closure technique (VCCT) [5,6], variational multiscale cohesive method (VMCM) [7], continuum decohesive finite element (CDFE) [8], and cohesive zone model (CZM) [9,10]. Among these methods, the CZM stands out as one of the most effective.

Initially, the CZMs were developed to describe the crack propagation under monotonic loading, with the traction-separation law exhibiting an elastic reversible behaviour [11]. The upper and

lower surfaces of a crack in the CZMs were assumed to be bonded together by the cohesive forces related to the surface displacements [12]. Subsequently, a few cohesive zone models were proposed to characterize the nonlinear behaviours of brittle materials at crack tips [13]. So far, a range of cohesive zone laws have been established, including linear [14], exponential [11] and trapezoidal models [15]. Recently, a strength-based cohesive zone model (ST-CZM) has been developed [16], redefining the traditional traction-separation law based on strength concept. However, the methodology for calibrating the parameters of this model remains imperfect.

Numerous current experimental methods are employed to characterize the fracture properties of interfaces [17–19]. Nevertheless, the experiments needed to determine these material constants are quite complex [20], requiring multiple tests to yield acceptable results. Therefore, accurately determining interface parameters through numerical or experimental methods continues to pose significant challenges.

In recent years, the inverse methods have increasingly attracted a lot of attention [20–22]. Wang et al. [23] proposed a method to identify the time-dependent interfacial mechanical properties. Cricrì [24] introduced a methodology capable of accurately identifying the cohesive law of adhesive layers under shear load. Valoroso et al. [25] developed an inverse identification method for Mode-I fracture interface parameters. Khosrozadeh et al. [20] integrated the finite element model (FEM) with the experiment models to identify the parameters of CZMs. However, these efforts in inverse parameter identification have predominantly focused on the conventional experimental models such as double cantilever beam (DCB) and fixed-ratio mixed-mode (FRMM), which are characterized by unidirectional crack propagation. Furthermore, the interface theories that describe the mechanical behaviour of cohesive elements are often based on several assumptions, which limits their applicability to real structures. Consequently, identifying interface characteristic parameters that align with the real properties of composite structures accurately remains a significant challenge [26]. A numerical method which can characterize the discontinuous lengths of debonding is essential for identifying the interface parameters in real structures.
Moreover, the advent and growth of artificial intelligence have drawn widespread attention. Genetic algorithms (GAs), a type of evolutionary computation technique, have been employed to solve various optimization problems [27]. As widely recognized intelligent optimization algorithms, GAs have evolved significantly since their appearance, being applied to solve a broad range of optimization challenges [28–30].

Employing GAs for inverse identification of interface parameters is an innovative approach [31–33]. However, the traditional genetic algorithms often become trapped in the local optima, and their use requires substantial computational resources. The adoption of a modified version, known as the multi-island genetic algorithm (MIGA), effectively mitigates these challenges inherent in the traditional GAs [34]. The MIGA has demonstrated its effectiveness across various fields, showing its robust global optimization abilities [35,36]. Compared with the traditional optimization approaches, the application of MIGA in nonlinear constrained optimization scenarios can avoid problems related to the local optima and complex sensitivity calculations [37].

In this context, the objective function is modified to represent the debonding lengths in real structures. Thus, the precision of the interface parameters obtained through the inverse method is enhanced. The developed objective function simultaneously considers the load-displacement curves and the debonding lengths in complex structures. This framework integrates the FEM and MIGA, and through an application programming interface (API), it facilitates the identification of parameters for any interface constitutive model which can be implemented in Abaqus. To validate the accuracy of the proposed method, a four-point bending model is introduced.

2. The cohesive element models

The CZMs are generally regarded as the effective methods for predicting the separation behaviours of composite materials. The mechanical behaviours of CZMs are described by constitutive laws. In this study, the recently proposed ST-CZM is utilized to reproduce the debonding mechanism of interfaces. The ST-CZM has been implemented in Abaqus through the development of a user-defined element (UEL) subroutine written in Fortran.

The constitutive law of ST-CZM can be formulated as

$$S_i(\omega_i)=\sigma_{i,c}\frac{\Delta u_{i,f}-\omega_i}{\Delta u_{i,f}-\Delta u_{i,c}},\quad (i=n,t) \tag{1}$$

Herein, the subscripts n and t denote the normal and tangential directions, respectively. $\sigma_{i,c}$ is the peak cohesive traction, while $k_{i,0}$ represents the initial elastic stiffness. Moreover, $\Delta u_{i,c}$ signifies the displacement gap at the onset of damage and $\Delta u_{i,f}$ corresponds to the displacement gap associated with traction vanishment. The bilinear constitutive curve is divided into two parts. The first considers the elastic stage ($\Delta u_i \le \Delta u_{i,c}$), where the relationship between the load and separation is linear elastic. The strength reaches its peak value at the end of first stage and the crack initials consequently. The strength finally decreases to zero as the crack propagates in the second part.

The damage scalars D_n and D_t in ST-CZM are formulated as

$$D_n=1-\frac{S_n(\omega_n)}{k_{n,0}\omega_n},\ D_t=1-\frac{S_t(\omega_t)}{k_{t,0}\omega_t} \tag{2}$$

The relation between interfacial traction and separation is characterized by constitutive curves (as illustrated in Fig.1). When considering the specific constitutive models, the key parameters such as $\sigma_{i,c}$, $\Delta u_{i,c}$, and $\Delta u_{i,f}$ vary independently. Thus, these parameters can effectively characterize the interfacial mechanical properties [26]. Utilizing different interface parameters will certainly lead to varying results in FEM. In this context, the design variable of the objective

function can be expressed as

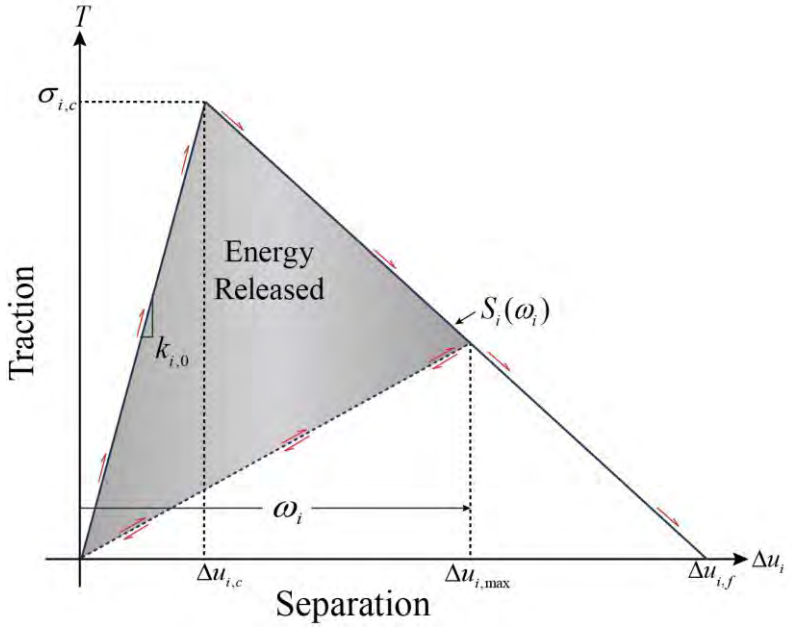


Fig.1 A bilinear cohesive traction-separation curve

$$\mathbf{x} = \left[\sigma_{n,c} \quad \Delta u_{n,f} \quad \Delta u_{n,c} \quad \sigma_{t,c} \quad \Delta u_{t,f} \quad \Delta u_{t,c} \right]^T \tag{3}$$

Since the fracture energy and initial elastic stiffness are closely linked to these parameters in the bilinear ST-CZM, they are not considered as independent target variables for calibration.

3. Introduction of identifying framework

The proposed identification framework integrates three interconnected modules: the FEM module, the objective function module, and the MIGA optimization module. The initial computational results are obtained by FEM, and the interface parameters in the model uses an initial guess value. The specific results, such as load-displacement curves and debonding length-displacement curves, are extracted to compute the objective function. The optimization is stopped once the calculated results satisfy the convergence criteria. If not, the value of objective function and simulation results will be used as references for subsequent steps. The flow chart of the framework is shown in Fig.2. Moreover, the efficiency of proposed framework is validated in our previous work [22]. In this paper, the objective function and identification method is modified to adopt the complexity of real experiment.

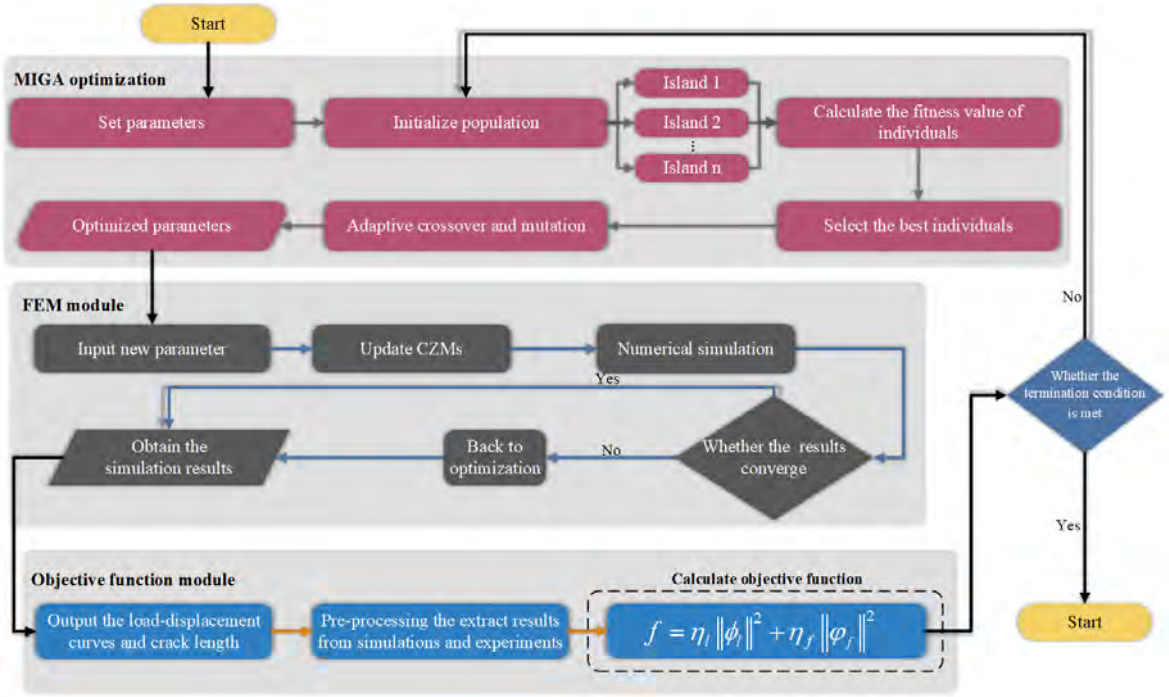


Fig.2 The flow chart of the proposed framework

3.1 Objective function

The majority of current inverse methods primarily focus on matching the load-displacement curves between numerical and experimental results [23,38,39]. In the inverse problems without strictly optimized boundary conditions, the accurately identified load-displacement curves cannot necessarily indicate the failure modes of the specimens have consistency. Therefore, it is essential to establish an objective function that considers the multiple factors. The previous inverse methods mostly use those experiments with unidirectionally propagating cracks as examples, which do not accurately reflect the real-world conditions. The objective function needs to be re-designed for considering the complex debonding conditions observed in actual structures.

To address the above-mentioned limitations in current inverse methods, a minimization function is necessary for numerical iteration. The nonlinear objective function, designed to quantify the differences between the experimental and numerical results, is defined as

$$f = \eta_l \|\phi_l\|^2 + \eta_f \|\varphi_f\|^2 \tag{4}$$

where the values of scalars η_l and η_f vary between 0 and 1, weighting the two parts of the function. ϕ_l and φ_f represent the objective function considering the debonding length and load-displacement curves, respectively. It should be noted here that the η_l and η_f are set to be 0.5 in our example.

Unlike the models featuring unidirectional crack propagation such as DCB and FRMM,

calibrating the debonding lengths at the fibre reinforced polymer (FRP)-concrete interface is challenging due to the randomness of debonding locations and directions. Therefore, it is necessary to adopt a universal method to formulate the complex debonding lengths of such interfaces. For internal elements that are difficult to observe, it is assumed that there are n elements along the considered interface in the numerical model. Each element is represented by a numerical subscript j ($j = 1, 2, 3, ..., n$). For simplicity, we assume that the failure of elements along the width direction in a three-dimensional model occurs simultaneously.

The element length in the finite element model is determined by the mesh density and represented by s. In finite element simulations, n_j is initially set to 0. When the damage scalar D equals 1, n_j is set to 1. The numerical debonding length of the interface is thus expressed as

$$\mathbf{N}_l = s * \Sigma n_j \tag{5}$$

where $\bar{\mathbf{N}}_l$ represents the debonding length obtained from experiments. Further, a function ϕ_l is proposed to express whether the simulated debonding length matches the experimental data, which is written as

$$\phi_l = \frac{\mathbf{N}_l - \bar{\mathbf{N}}_l}{\left\| \bar{\mathbf{N}}_l \right\|} \tag{6}$$

The objective function φ_F can be expressed as

$$\varphi_f = \frac{\mathbf{M}_f - \bar{\mathbf{M}}_f}{\left\| \bar{\mathbf{M}}_f \right\|} = \frac{\mathbf{r}_f}{\left\| \bar{\mathbf{M}}_f \right\|} \tag{7}$$

where

$$\mathbf{M}_f = \begin{bmatrix} M_1 & M_2 & \dots & M_k \end{bmatrix},\ \bar{\mathbf{M}}_f = \begin{bmatrix} \bar{M}_1 & \bar{M}_2 & \dots & \bar{M}_k \end{bmatrix} \tag{8}$$

and the residual vector $\mathbf{r}_f$ is given by

$$\mathbf{r}_f = \begin{Bmatrix} M_1 - \bar{M}_1 \\ M_2 - \bar{M}_2 \\ \vdots \\ M_k - \bar{M}_k \end{Bmatrix} \tag{9}$$

Herein, M_k represents the load obtained from simulation, and $\bar{M}_k$ is the load obtained from experiments. To simplify the objective function calculation, both numerical and experimental data are processed using the interpolation techniques to ensure one-to-one correspondence in the results. Furthermore, the subscript k here denotes the discrete data points of the interpolation results.

3.2 Optimization algorithm

Since the global convexity of objective function $f(x)$ is difficult to be evaluated [21], the metaheuristics approaches such as Simulated Annealing [40] and Genetic Algorithms [41] are proposed to optimize this type of problems.

The multi-island genetic algorithm used in this study is an enhancement of Parallel Distributed Genetic Algorithms (PDGAs) [37]. Compared to the traditional GAs, the MIGA offers the improved computational efficiency and can avoid getting trapped in local optima. The MIGA divides the entire population into multiple islands as depicted in Fig.2. Within each island, subpopulations are selected independently through the traditional GA processes including selection, crossover, mutation, and so on. An elite retention strategy ensures the transmission of the best genes to subsequent generations. Furthermore, the MIGA randomly selects individuals for migration operation to maintain the population diversity. This approach ensures that each island receives the new genetic information, thereby preventing the results from becoming trapped in local optima and inhibiting the premature maturity.

4. Numerical results and validation

The debonding failure modes of traditional validation models such as DCB and SLB tests are relatively straightforward [25,42,43]. To validate that the parameters derived from the proposed inverse framework can faithfully reproduce the failure mode of experiment, this section provides another verification using a complex experimental model (i.e. the four-point bending test) [44]. Different from the DCB test, various debonding failure modes occur in the four-point bending test, including plate end interfacial debonding, critical diagonal crack-induced (CDC) debonding, and intermediate flexural crack (IC) induced interfacial debonding, which further lead to the complex failure patterns in the concrete [44].

The geometric properties and boundary conditions of the test are shown in Fig.3. The steel bars are embedded in the concrete. The compressive strength of the concrete is 47 MPa, and the large-span concrete beam measures a length of $L_{beam}=4000\text{mm}$, a width of $w_{beam}=200\text{mm}$, and a height of $H=450\text{mm}$. The reinforced bars are detailed in Fig.3. The FRP plate, adhered to the bottom, has a length of $L_{FRP}=3800\text{mm}$, a width of $w_{FRP}=100\text{mm}$, and a thickness of $t_{FRP}=0.999\text{mm}$.

In addition, a denotes the distance between two loading points. In four-point test-A, $a=1250\text{mm}$. A velocity of 0.1m/s is applied to the loading position, the concrete is meshed by C3D8 element, and the steel is embedded in the model. ST-CZM is implemented using a UEL in Abaqus.

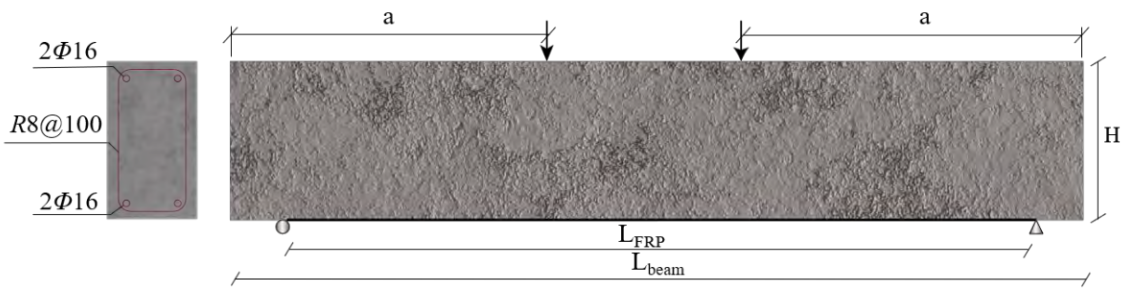


Fig.3 Geometry and boundary conditions of the four-point test specimen

In the experiment [44], the value of the debonding length across the complete process is not provided. However, the debonding length corresponding to both initial and final debonding is given, which can also form the objective function. For this experiment, the objective function is simplified to $\mathbf{r}_L(\boldsymbol{x}) = \begin{pmatrix} L_1^{\exp} - L_1^{nu}(\boldsymbol{x}) \\ L_{end}^{\exp} - L_{end}^{nu}(\boldsymbol{x}) \end{pmatrix}$ without intermediate variables.

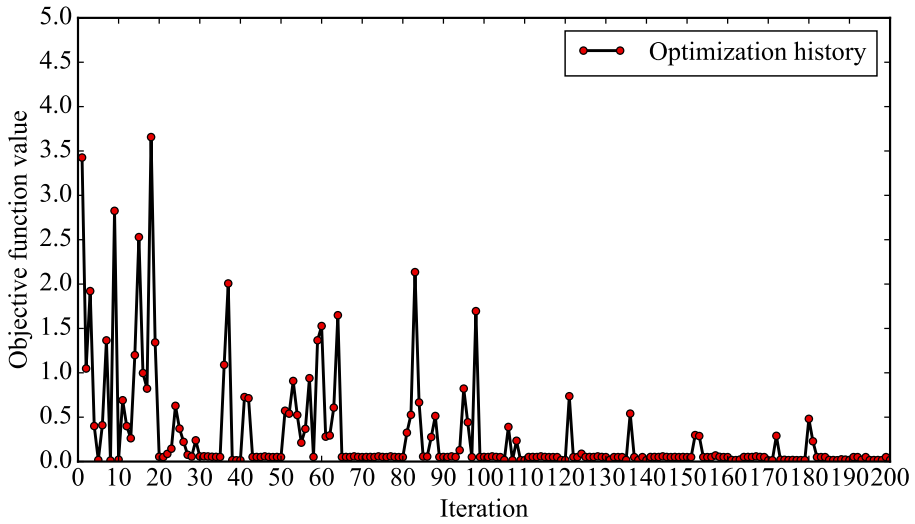


Fig.4 Iteration process for the four-point test-A

The debonding length is estimated to be 436mm at the onset of local debonding [44]. At the end of the experiment, the strength contribution from the FRP plate has nearly vanished, with the final debonding length approximated at 1048mm [44]. To compare the simulation results with the experimental data, the load is converted to the mid-span moment. This adjustment does not affect the objective function. Fig.4 depicts the iteration process during the optimization, in which the objective function eventually converges around zero. Table 1 details the interface parameters derived from the proposed framework. Fig.5 indicates the load-displacement curve for this simulation. It can be found that even though the objective function is simplified, the results are in the acceptable range.

Table 1 Results from inverse parameter identification for the four-point test-A

	$\sigma_{n,c}$ [MPa]	$\sigma_{t,c}$ [MPa]	$\Delta u_{n,c}$ [mm]	$\Delta u_{t,c}$ [mm]	$\Delta u_{n,f}$ [mm]	$\Delta u_{t,f}$ [mm]
Cohesive Element	12	14	0.01153	0.01297	0.1457	0.1504
	$k_{n,0}$ [N/mm^3]	G_I [N/mm]	$k_{t,0}$ [N/mm^3]	G_{II} [N/mm]		
	1040.76	0.8742	1079.41	1.0528		

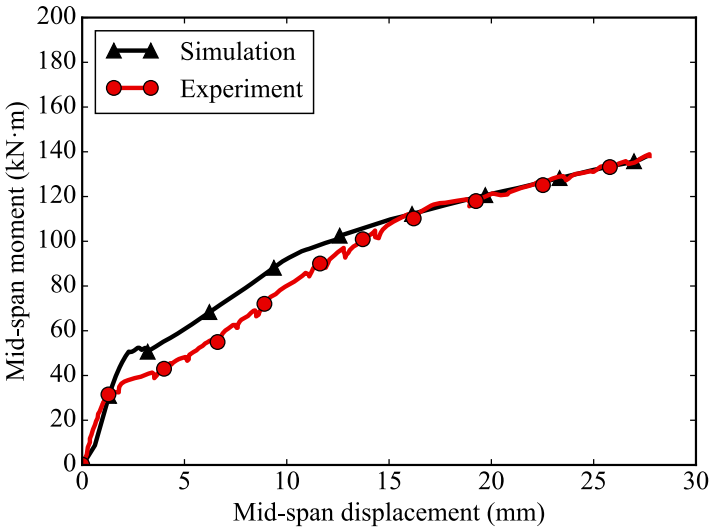


Fig.5 Moment-deflection curves for the four-point test-A

Fig.6 illustrates the failure mode of experiment, while Fig.7 presents the failure mode obtained by simulation. It is found that the numerical results, derived from the optimal parameters, closely mirror the failure mode observed in the experiments. The cracks initiate between the loading points and develop upwards, while other cracks progress towards the loading positions.

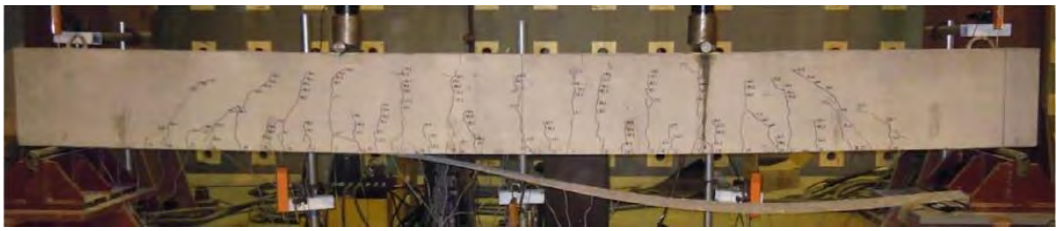


Fig.6 Failure mode of experiment for the four-point test-A [44]

Fig.8 displays the diagram of local debonding marks from the simulation. The result is obtained from the calculation step which represents the debonding initiation. The damage distribution map is derived using the average value along the thickness direction (as shown in Fig.8 (b)). It is revealed that the numerically predicted zones of concentrated damage are close to the local debonding marked areas of experiment.

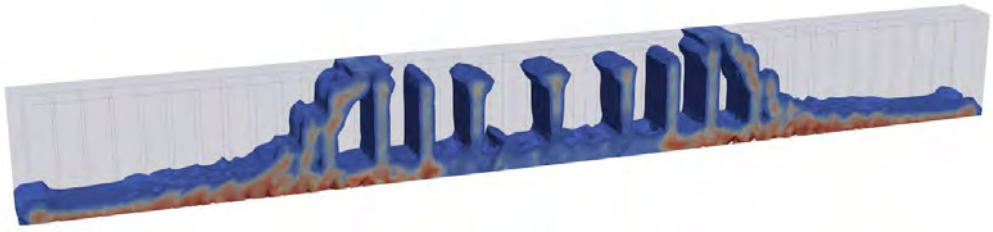


Fig.7 Failure mode obtained from simulation for the four-point test-A

To further illustrate the applicability of the interface parameters obtained from the proposed framework in similar models, another four-point test is simulated (named the four-point test-B). Compared to the four-point test-A, the new model (i.e. four-point test-B) retains the same material properties and geometry conditions, while a=1750mm [44]. Moreover, the key parameters are set to be the results obtained from four-point test-A (as shown in Table 1). The moment-displacement curves are shown in Fig.9. As can be seen, the simulation results reproduce the experimental results.

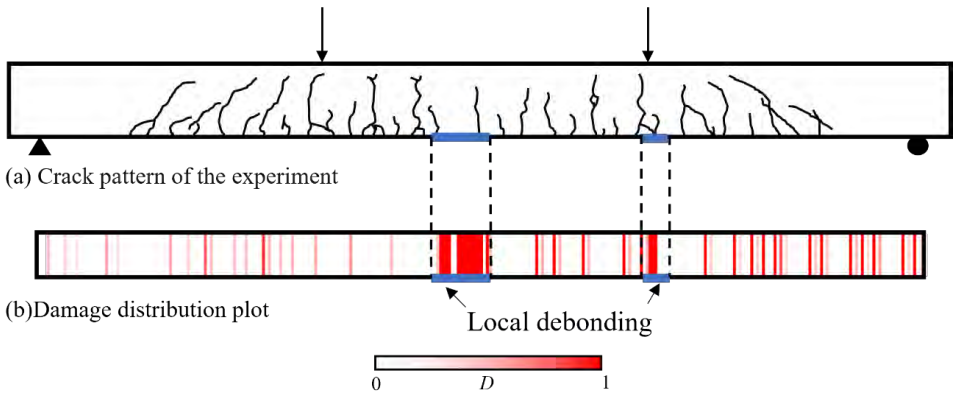


Fig.8 Crack pattern of the experiment and damage distribution plot of cohesive element for the four-point test-A

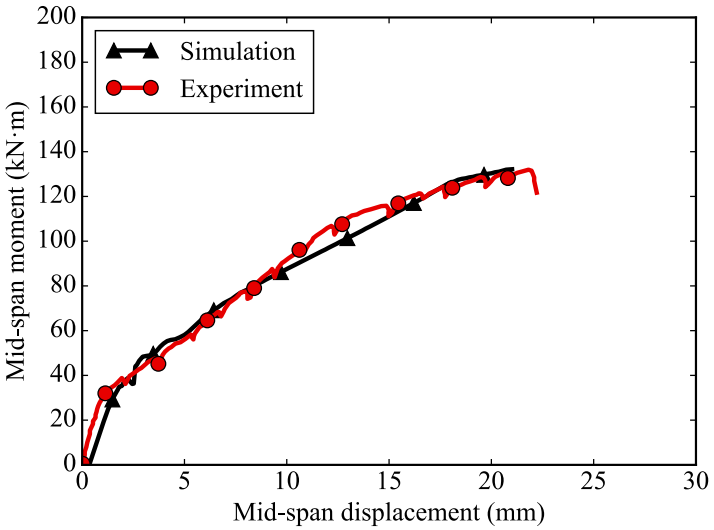


Fig.9 Relationship curves between mid-span moment and mid-span displacement for the four-point test-B

The crack development in concrete, which is obtained from the simulation and shown in Fig.11, displays the results similar to the experimental observations (as shown in Fig.10). The cracks between the two loading points tend to develop upwards, while those outside this area extend towards the loading points. Furthermore, the interface parameters successfully predict the local debonding locations at the concrete-FRP interface (as illustrated in Fig.12).

Fig.10 Failure mode of experiment for the four-point test-B [44]

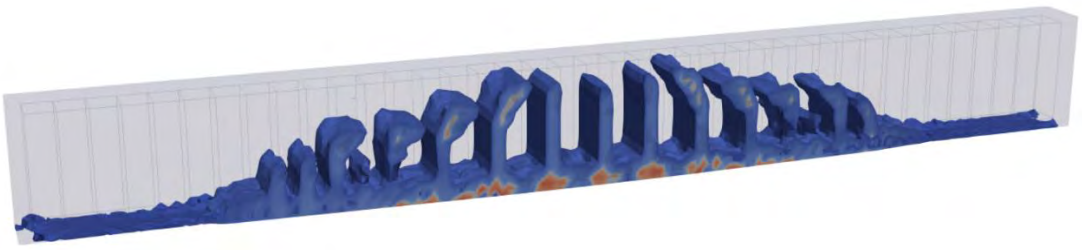


Fig.11 Failure mode obtained from simulation for the four-point test-B

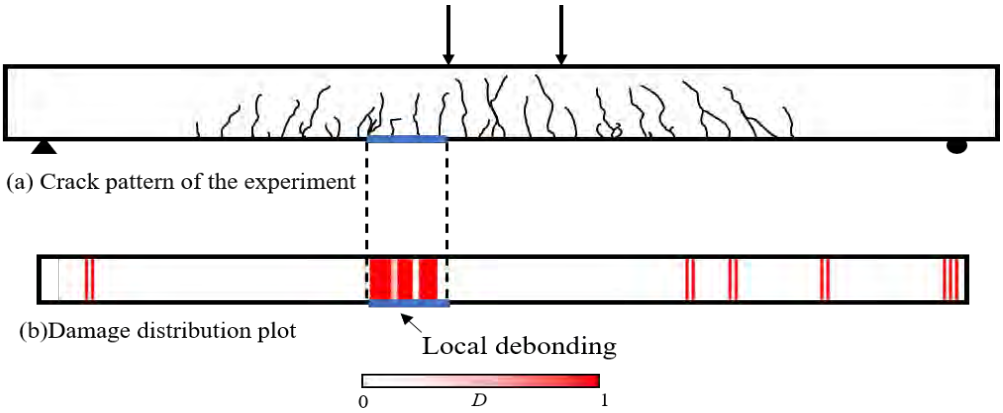


Fig.12 Crack pattern of the experiment and damage distribution plot of cohesive element for the four-point test-B

Being a representative engineering experiment, IC debonding in the full-scale four-point beams is closely related to the distribution of interfacial shear stress [44]. It is particularly meaningful using the four-point bending test as an example to validate the proposed inverse framework. Moreover, the interface parameters derived from the four-point bending test are applicable in a wide range of engineering cases [44,45].

5. Conclusion and discussion

This study proposes a method to extract the debonding lengths from the numerical model. The method considers the complex debonding conditions observed in real experiments. A nonlinear multi-objective function which takes the load-displacement curve and crack length as input data are presented to formulate the identification problem. A numerical model that reproduces the four-point bending test is established, implementing interface elements based on ST-CZM. In this context, the interface parameters of the FRP-concrete in four-point bending model are accurately identified using an inverse identification framework based on MIGA. The identified parameters accurately reproduce the failure modes observed in the experiments, thus verifying the accuracy and effectiveness of the proposed method.

CRediT authorship contribution statement

Tianxiang Shi: Methodology, Investigation, Validation, Writing original draft, Simulation. **Yangyang Wang:** Data curation. **Miao Pang:** Investigation, Data curation. **Yongqiang Zhang:** Writing review and editing, Supervision, Project administration.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

References

[1] A. Taherzadeh-Fard, A. Cornejo, S. Jiménez, L.G. Barbu, A rule of mixtures approach for delamination damage analysis in composite materials, Composites Science and Technology 242 (2023) 110160.

[2] A. Mahmoudi, M.R. Khosravani, M.M. Khonsari, T. Reinicke, On the evaluation of entropy threshold for debonding during crack prorogation using DIC technique, Engineering Fracture Mechanics 288 (2023) 109361.

[3] T. Shi, Y. Zhang, X. Zhang, Y. Wang, K. Zheng, A strength based thermo-mechanical coupled cohesive zone model for simulating heat flux induced interface debonding, Composites Science and Technology 243 (2023) 110255.

[4] S. Wang, D. Li, Z. Li, J. Liu, S. Gong, G. Li, A rate-dependent model and its user subroutine for cohesive element method to investigate propagation and branching behavior of dynamic brittle crack, Computers and Geotechnics 136 (2021) 104233.

[5] P.S. Valvo, A Physically Consistent Virtual Crack Closure Technique for I/II/III Mixed-mode Fracture Problems, Procedia Materials Science 3 (2014) 1983–1987.

[6] R. Krueger, Virtual crack closure technique: History, approach, and applications, Applied Mechanics Reviews 57 (2004) 109–143.

[7] J. Mergheim, A variational multiscale method to model crack propagation at finite strains, International Journal for Numerical Methods in Engineering 80 (2009) 269–289.

[8] P. Prabhakar, A.M. Waas, A novel continuum-decohesive finite element for modeling in-plane fracture in fiber reinforced composites, Composites Science and Technology 83 (2013) 1–10.

[9] X. Lu, M. Ridha, B.Y. Chen, V.B.C. Tan, T.E. Tay, On cohesive element parameters and delamination modelling, Engineering Fracture Mechanics 206 (2019) 278–296.

[10] H.C. Hyun, F. Rickhey, J.H. Lee, M. Kim, H. Lee, Evaluation of indentation fracture toughness for brittle materials based on the cohesive zone finite element method, Engineering Fracture Mechanics 134 (2015) 304–316.

[11] X -P Xu, A Needleman, Void nucleation by inclusion debonding in a crystal matrix, Modelling and Simulation in Materials Science and Engineering 1 (1993) 111.

[12] A. Needleman, Micromechanical modelling of interfacial decohesion, Ultramicroscopy 40 (1992) 203–214.

[13] G.I. Barenblatt, The Mathematical Theory of Equilibrium Cracks in Brittle Fracture, in: H.L. Dryden, Th. von Kármán, G. Kuerti, F.H. van den Dungen, L. Howarth (Eds.), Advances in Applied Mechanics, Elsevier, 1962: pp. 55–129.

[14] A. Turon, C.G. Dávila, P.P. Camanho, J. Costa, An engineering solution for mesh size effects in the simulation of delamination using cohesive zone models, Engineering Fracture Mechanics 74 (2007) 1665–1682.

[15] R.M.R.P. Fernandes, J.A.G. Chousal, M.F.S.F. de Moura, J. Xavier, Determination of cohesive laws of composite bonded joints under mode II loading, Composites Part B: Engineering 52 (2013) 269–274.

[16] P.F. Liu, M.M. Islam, A nonlinear cohesive model for mixed-mode delamination of composite laminates, Composite Structures 106 (2013) 47–56.

[17] C. Morin, S. Avril, Inverse problems in the mechanical characterization of elastic arteries, MRS Bulletin 40 (2015) 317–323.

[18] S. Guessasma, D.H. Bassir, Identification of mechanical properties of biopolymer composites sensitive to interface effect using hybrid approach, Mechanics of Materials 42 (2010) 344–353.

[19] A. Khosrozadeh, A. Khosravifard, I. Rajabi, Inverse identification of material constants of various cohesive laws for delamination of composites using experimental results, Composite Structures 303 (2023) 116241.

[20] C. Chisari, L. Macorini, C. Amadio, B.A. Izzuddin, An inverse analysis procedure for material parameter identification of mortar joints in unreinforced masonry, Computers & Structures 155 (2015) 97–105.

[21] T. Shi, M. Pang, Y. Wang, Y. Zhang, Inverse parameter identification framework for cohesive zone models based on multi-island genetic algorithm, Engineering Fracture Mechanics (2024) 110005.

[22] J. Wang, Y.L. Kang, Q.H. Qin, D.H. Fu, X.Q. Li, Identification of time-dependent interfacial mechanical properties of adhesive by hybrid/inverse method, Computational Materials Science 43 (2008) 1160–1164.

[23] G. Cricrì, Cohesive law identification of adhesive layers subject to shear load – An exact inverse solution, International Journal of Solids and Structures 158 (2019) 150–164.

[24] N. Valoroso, S. Sessa, M. Lepore, G. Cricrì, Identification of mode-I cohesive parameters for bonded interfaces based on DCB test, Engineering Fracture Mechanics 104 (2013) 56–79.

[25] Y.L. Kang, X.H. Lin, Q.H. Qin, Inverse/genetic method and its application in identification of mechanical parameters of interface in composite, Composite Structures 66 (2004) 449–458.

[26] M. Drissi El-Bouzaidi, T. El Bardouni, E. Chham, O. El Hajjaji, A. Nouayti, A. Idrissi, K. El-Bakkari, Unfolding of the neutron spectrum in a nuclear reactor using genetic algorithms method, Nuclear Engineering and Design 422 (2024) 113156.

[27] R.R. Bies, M.F. Muldoon, B.G. Pollock, S. Manuck, G. Smith, M.E. Sale, A Genetic Algorithm-Based, Hybrid Machine Learning Approach to Model Selection, Journal of Pharmacokinetics and Pharmacodynamics 33 (2006) 195–221.

[28] P.A. Wrigley, P. Wood, P. Stewart, R. Hall, D. Robertson, Module layout optimization using a genetic algorithm in light water modular nuclear reactor power plants, Nuclear Engineering and Design 341 (2019) 100–111.
[29] Z. Zhao, Y. Bao, T. Gao, Q. An, Optimization of GFRP-concrete-steel composite column based on genetic algorithm - artificial neural network, Applied Ocean Research 143 (2024) 103881.
[30] X.H. Lin, Y.L. Kang, Q.H. Qin, D.H. Fu, Identification of interfacial parameters in a particle reinforced metal matrix composite Al6061–10%Al2O3 by hybrid method and genetic algorithm, Computational Materials Science 32 (2005) 47–56.
[31] G.-C. Luh, C.-Y. Wu, Inversion control of non-linear systems with an inverse NARX model identified using genetic algorithms, Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering 214 (2000) 259–271.
[32] P. Liu, Y. Xie, S.-J. Guo, G.-P. Cai, Identification Technology of Modal Parameter of a Strap-on Launch Vehicle, Int. J. Comput. Methods 13 (2016) 1650010.
[33] X. Hu, X. Chen, Y. Zhao, W. Yao, Optimization design of satellite separation systems based on Multi-Island Genetic Algorithm, Advances in Space Research 53 (2014) 870–876.
[34] D.J. Zhao, Y.K. Wang, W.W. Cao, P. Zhou, Optimization of Suction Control on an Airfoil Using Multi-island Genetic Algorithm, Procedia Engineering 99 (2015) 696–702.
[35] R. Lü, X. Guan, X. Li, I. Hwang, A large-scale flight multi-objective assignment approach based on multi-island parallel evolution algorithm with cooperative coevolutionary, Science China Information Sciences 59 (2016) 072201.
[36] K. Zhang, H. Ma, Q. Li, D. Wang, Q. Song, X. Wang, X. Kong, Thermodynamic analysis and optimization of variable effect absorption refrigeration system using multi-island genetic algorithm, Energy Reports (2022).
[37] J.A. Nairn, Y.E. Aimene, A re-evaluation of mixed-mode cohesive zone modeling based on strength concepts instead of traction laws, Engineering Fracture Mechanics 248 (2021) 107704.
[38] S.M. Jensen, M.J. Martos, E. Lindgaard, B.L.V. Bak, Inverse parameter identification of n-segmented multilinear cohesive laws using parametric finite element modeling, Composite Structures 225 (2019)
[39] N. Antoni, An inverse method for the identification of parameters in beam contact problems, Journal of Computational Science 72 (2023) 102111.
[40] S. Kirkpatrick, C.D. Gelatt, M.P. Vecchi, Optimization by Simulated Annealing, Science 220 (1983) 671–680.
[41] D. Chen, Y. Li, X. Yang, W. Jiang, L. Guan, Efficient parameters identification of a modified GTN model of ductile fracture using machine learning, Engineering Fracture Mechanics 245 (2021) 107535.
[42] F.A.M. Pereira, J.J.L. Morais, M.F.S.F. de Moura, N. Dourado, M.I.R. Dias, Evaluation of bone cohesive laws using an inverse method applied to the DCB test, Engineering Fracture Mechanics 96 (2012) 724–736.
[43] F.A.M. Pereira, M.F.S.F. de Moura, N. Dourado, J.J.L. Morais, J. Xavier, M.I.R. Dias, Direct and inverse methods applied to the determination of mode I cohesive law of bovine cortical bone using the DCB test, International Journal of Solids and Structures 128 (2017) 210–220.
[44] B. Fu, J.G. Teng, G.M. Chen, J.F. Chen, Y.C. Guo, Effect of load distribution on IC debonding in FRP-strengthened RC beams: Full-scale experiments, Composite Structures 188 (2018) 483–496.
[45] J. Yao, J.G. Teng, J.F. Chen, Experimental study on FRP-to-concrete bonded joints, Composites Part B: Engineering 36 (2005) 99–113.

GPU-based parallel WCSPH method for free surface flows

***Xi Yang[1], Guangqi Liang[1], Taian Hu[3], †Guiyong Zhang[1,2], Zhifan Zhang[1]**

[1]School of Naval Architecture Engineering, Dalian University of Technology, Dalian, China
[2]Collaborative Innovation Center for Advanced Ship and Deep-Sea Exploration, Shanghai, China
[3]Marine Design and Research Institute of China, Shanghai, China

*Presenting author: xiyang_sph@163.com
†Corresponding author: gyzhang@dlut.edu.cn

Abstract

In this work, a GPU-based parallel WCSPH framework is constructed based on the CUDA programming language. Except for the pre-process and post-process running on the CPU, all the SPH calculations are implemented on the GPU. The speedup analysis of the present GPU code compared with the serial CPU one is conducted by a water tank test. The maximum acceleration ratio is over 100. The developed GPU code is firstly validated by a hydrostatic benchmark. The results agree well with the analytical solution, the accuracy and convergence are verified. Then, the case of 3D dam breaking flow impacting the rigid obstacle is simulated to show the ability of the current parallel model. The simulation results are in good agreement with those of the experiments.

Keywords: GPU parallelization, CUDA, SPH

1. Introduction

Particle methods have incomparable advantages in dealing with free surface or interface flows due to the Lagrangian description and particle discretization. Smoothed particle hydrodynamics (SPH) is one of the most popular meshless particle methods. It is originally proposed for the astrophysics to study the celestial mechanics [1][2]. After Monaghan [3] introduced it into the hydrodynamics, SPH shows extraordinary superiority in simulating free surface flows. The stability, accuracy and robustness of SPH method have been greatly enhanced in recent years. Various numerical techniques are developed to improve it [4]-[7]. The SPH is successfully applied in ocean engineering, solving green water impacting, wave interaction, structure slamming, sloshing and so on. Although the SPH method has achieve enormous progresses in computational fluid dynamics, there are still some challenging issues existing, like the boundary condition enforcement and the computational efficiency. The huge computational cost is the main factor restricting the application of SPH in simulating the real-scale engineering problems. The time-dependent topological relationships between the interaction particles need to be detected in every time step, inducing large computational expense. With the increasement of the particle amount, the solution cost will be unbearable.

There are two known methods to lower the computational cost and boost efficiency. The first one is to implement multi-resolution algorithm to save the computational resources. The adaptive particle refinement (APR) like adaptive mesh refinement (AMR) in mesh-based method has been developed [7]. In this manner, only the target area of the computational domain is solved by high-resolution particles, the remaining portion is discretized by coarse particles. To some extent, the trade-off between computational cost and efficiency is met [8].

The overall effect of such measure is limited. Besides, calculation errors are easily introduced and accumulate due to the intrinsic defect of this multi-resolution algorithm. Another way to improve the computational efficiency is to use parallel programming. The compute unified device architecture (CUDA) is a parallel computing platform and programming model invented by NVIDIA [9]. It provides a significant increase in computational performance by utilizing the processing power of the graphics processing unit (GPU). As is shown in Fig. 1, CPU has complex internal structure containing controllers, caches and cores, where the core is the main unit to work. CPU preforms well when the instructions with complex logic are given. Commonly used personal computers generally own a few cores, and the work station may have hundreds of cores. GPUs have a more straightforward internal structure with fewer controllers and more processing units, making them ideal for large-scale scientific computing. The number of cores on common GPU devices is hundreds to tens of thousands, significantly more than on CPUs. Despite the fact that a single GPU core cannot compute as quickly as a CPU, the great acceleration effect can still be achieved with a significant number of cores. By converting CPU serial programs to GPU parallel programs, several researchers have used the GPU to accelerate SPH computing.

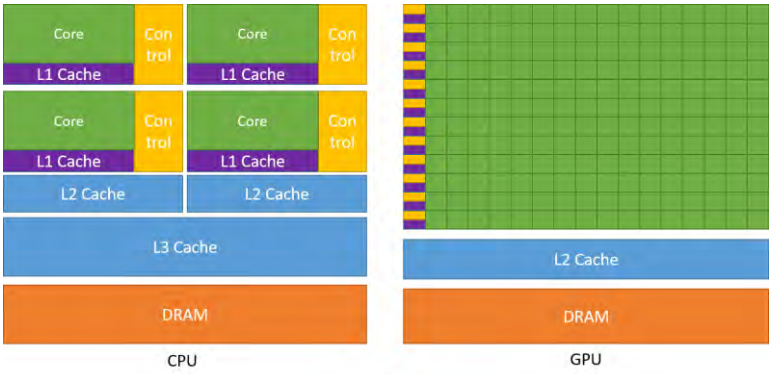


Figure 1. Illustrations of CPU and GPU architectures [9]

Harada et al. [10] and Zhang et al. [11] are the pioneers in using SPH by GPU programming. The well-known open-source DualSPHysics uses CUDA C++ to implement parallelism [12]. The parallel power computing of GPU is used to accelerate DualSPHysics by up to two orders of magnitude compared to the performance of the serial version [13]. A GPU accelerated SPH method is developed by Zhang et al. [14] using CUDA programming for water entry of a 3D circular cylinder. Nearly ten million of particles are generated for the simulation and hundreds of speedups is achieved. Zhao et al. have developed a new high-efficiency SPH algorithm with a multi-cell linked list coupled with adaptive particle refinement. Their SPH code is implemented with a CUDA library base on a GPU [15]. Based on this model, Zhao et al. proposed a multi-phase, multi-resolution SPH method for fluid/solid interaction with multi-GPU implementation and dynamic load balancing following the movement of the refinement regions [16]. The simulation of high-speed water entry is realized with more than 100 million particles. A graphical processing unit implementation of adaptive particle refinement in the SPH framework is also proposed by Yang et al. [17] to simulate the multi-phase flow and fluid-structure coupling problems. Zhan et al. [18] developed the full-particle GPU model based on the total Lagrangian SPH and WCSPH and applied on the 3D fluid structure interactions, the computational accuracy and efficiency are verified. A GPU-based coupled DEM-SPH model is proposed by Zhang et al. [19] and adopted to simulate dike failure problems. Mao et al. [20] investigated the impact of GPU parallel architecture, low-precision computing on GPUs, and GPU memory management on NNPS efficiency. They employed a GPU-accelerated mixed-precision SPH framework, utilizing low-precision float-point 16 (FP16) for NNPS while maintaining high precision for other components. The coupling

CUDA and MPI acceleration modes are proposed by Chen et al. [21], which exploit the high computational capacity of GPU devices and the cross-device parallelism properties in multi-GPUs.

In this work, the parallelization of WCSPH is achieved by the CUDA toolkit based on the GPU developed by NVIDIA. The basic theory of SPH is simply reviewed at first, then the CUDA programming is introduced. The implementation of GPU parallel SPH is described in detail. After that, the computational efficiency enhancement is discussed. Based on the present parallel SPH model, the hydrostatic water is simulated to verify it. The results agree well with the analytical solution, the accuracy and convergence are validated. Then, the 3D dam breaking flow impacting the rigid obstacle are used to show the ability of the current parallel model in handling large scale simulation. Further, the present GPU SPH model is extended and coupled with the SPIM method to solve the FSI problem.

2. Basic SPH theory and GPU parallelization

2.1 SPH method

In this section, the delta-SPH model is introduced. The discretized governing equations are as follows [8]:

$$\begin{cases} \dfrac{d\rho_i}{dt} = \rho_i \sum_j (\mathbf{v}_i - \mathbf{v}_j)\nabla_i W_{ij} V_j + D_i \\ \dfrac{d\mathbf{v}_i}{dt} = -\dfrac{1}{\rho_i}\sum_j (p_i + p_j)\nabla_i W_{ij} V_j + \mathbf{g} + \mathbf{F}_i^v, \\ p_i = c_s^2(\rho_i - \rho_0) \end{cases} \tag{1}$$

The mass and momentum equations are solved by the particle interactions between neighbor particles. ρ denotes the fluid density, $\mathbf{v}$ is the velocity, p represents the pressure, V is the volume of the particle, c_s is the artificial sound speed. $\mathbf{g}$ and $\mathbf{F}^v$ are respectively the gravity and the artificial force [8]:

$$\begin{cases} \mathbf{F}_i^v = \dfrac{\alpha h c_s}{\bar{\rho}} \sum_j m_j \prod \nabla_i W_{ij} \\ \prod = \dfrac{(\mathbf{r}_i - \mathbf{r}_j)(\mathbf{v}_i - \mathbf{v}_j)}{|\mathbf{r}_i - \mathbf{r}_j|^2 + \eta^2} \end{cases} \tag{2}$$

The cubic spline kernel function W is used [23], with the smoothing length $h = 1.2dx$ and supporting radius $kh = 2h$:

$$W(q) = \frac{5}{14\pi h^2} \begin{cases} 3q^3 - 6q^2 + 1 & ,0 \le q < 1 \\ (2-q)^3 & ,1 \le q < 2, q = |\mathbf{r}_i - \mathbf{r}_j|/h \\ 0 & ,q \ge 2 \end{cases} \tag{3}$$

The delta term D_i to diffusive the density oscillations are given by [24]:

$$\begin{cases} D_i = \delta c_s h \sum_j \varphi \nabla_i W_{ij} V_j \\ \varphi = \dfrac{\mathbf{r}_i - \mathbf{r}_j}{|\mathbf{r}_i - \mathbf{r}_j|^2 + \eta^2}(\rho_i - \rho_j) \end{cases} \tag{4}$$

where δ is the tuned parameter, usually set as 0.1. $\eta = 0.1h$ to avoid singularity.

The fixed ghost particle is utilized to enforce the wall boundary condition. It is convenient and efficient in dealing with complex geometry shape. At the pre-process stage, the wall is

discretized by several layers of dummy particles. Their motion is governed by the pre-defined discipline. The boundary particles obtain the pressure by the following equation [25]:

$$p_w = \frac{\sum_f p_f W_{wf} + (\mathbf{g} - \mathbf{a}_w)\sum_f \rho_f (\mathbf{r}_w - \mathbf{r}_f) W_{wf}}{\sum_f W_{wf}}, \tag{5}$$

where the subscripts w and f respectively represent the wall particles and fluid particles. Then the density of the boundary particle is calculated by inverse the equation of state, and the mass is kept constant.

The prediction-correction scheme is utilized to solve the above equations. The first step is prediction step, the variables of the particles are updated by half of the time step. Then, the SPH calculation is implemented based on the updated information. The second step is correction step, the variables are updated according to the new time derivatives. In the third step, the variables of next time step are updated, the solution of one discrete time is achieved. The size of the time step is chosen by Eq. (7) to keep the stability of the simulation.

$$\begin{aligned}
Step1: &\begin{cases} \mathbf{v}_{n+1/2} = \mathbf{v}_n + \mathbf{a}_n \cdot dt/2 \\ \rho_{n+1/2} = \rho_n + \rho'_n \cdot dt/2 \\ \mathbf{r}_{n+1/2} = \mathbf{r}_n + \mathbf{v}_n \cdot dt/2 \end{cases} \\
Step2: &\begin{cases} \mathbf{v}_{n+1/2} = \mathbf{v}_n + \mathbf{a}_{n+1/2} \cdot dt/2 \\ \rho_{n+1/2} = \rho_n + \rho'_{n+1/2} \cdot dt/2 \\ \mathbf{r}_{n+1/2} = \mathbf{r}_n + \mathbf{v}_{n+1/2} \cdot dt/2 \end{cases} \\
Step3: &\begin{cases} \mathbf{v}_{n+1} = 2\mathbf{v}_{n+1/2} - \mathbf{v}_n \\ \rho_{n+1} = 2\rho_{n+1/2} - \rho_n \\ \mathbf{r}_{n+1} = 2\mathbf{r}_{n+1/2} - \mathbf{r}_n \end{cases}
\end{aligned} \tag{6}$$

$$dt \le \begin{cases} 0.25h/c_s \\ 0.25\min(h/|\mathbf{a}|)^{0.5} \end{cases} \tag{7}$$

2.2 GPU parallelization

The CPU serial procedure of the SPH simulation is first recalled. In a serial program, all particles and the particle pairs are the basis of the simulation, which is achieved by procedure-oriented programming. As is shown in Fig. 2, the solution of the governing equations is obtained by looping through all particles and particle pairs. The search for particle pairs and the calculation depending on the pairs are the most time-consuming parts of the solving process. The CPU core needs to traverse all particles and evaluate the relative distance to determine whether they are close enough to form the pair. It is a simple procedure but a huge amount of computation. After the nearest neighbor particles search, the interactions of inter-particles are solved to obtain the material derivatives. Taking the kernel function used in this work as an example, the proximity particles in the supporting domain of a single particle are about 30, so the particle pairs are about 30 times the number of particles, which is an order of magnitude higher than the time needed to simply loop through all particles. CPU can also fulfill the accelerated computation by using multiple threads tools. However, the extra overhead of the multi-threading process also increases the computational cost, and the number of CPU cores is limited.

```
for i=1:np                    for i=1:niac
   loop all particles...         loop all particle pairs....
end for                       end
```

Figure 2. Basic calculation of CPU SPH

GPU can have up to two orders of magnitude more computational cores than CPU, enabling enormous scientific computing. In this work, all SPH calculations are particle-associated and are programmed in object-oriented CUDA C++. In the GPU model, all particles separately evolve by the corresponding cores, and the whole process of the SPH method is implemented on a single particle. Here, the basic flow of the CUDA programming model is described:
Step 1: Allocate host memory and initialize data;
Step 2: Allocate device memory and copy data from host to device;
Step 3: Call CUDA kernel functions to complete the specified operations on the device;
Step 4: Copy the result on the device to the host;
Step 5: Release the memory allocated on the device and host.
In the CUDA model, the CPU is referred to as the host side and the GPU is referred to as the device side. A function called on the host and runes on the device is referred to as the *kernel function* (which needs to be differentiated from the kernel functions of the SPH method). The computational data is first initialized on the host and then calls the *cudaMalloc* function on the host to allocate a copy of the memory space on the device, followed by a call to *cudaMemcpy* to copy the data from the host to the device. After this, the GPU *kernel function* is called on the host to perform the computation. When the output is required, *cudaMemcpy* is used again to copy the data from the device to the host. At the end of the computation, the memory used on the host and device is released.

The GPU *kernel function* running on the device is the key to achieving the computation. In order to improve the cache hit rate of the particle pair search process to save computational cost, it is necessary to reorder the particles according to the background cell positions where the particles are located. The sorted particle is surrounded by the particles that will interact with it in the future. Assuming that the position of particle i is $x(i)$, $y(i)$ and $z(i)$, its background grid position, named *Hash* value, $Hash(i)$ is:

$$\begin{aligned} &Hash(i) = k + l \times K + m \times K \times L, \\ &(k,l,m) = (\left\lfloor \frac{x(i)-x_{\min}}{kh} \right\rfloor, \left\lfloor \frac{y(i)-y_{\min}}{kh} \right\rfloor, \left\lfloor \frac{z(i)-z_{\min}}{kh} \right\rfloor), \\ &K = \left\lfloor \frac{x_{\max}-x_{\min}}{kh} \right\rfloor, L = \left\lfloor \frac{y_{\max}-y_{\min}}{kh} \right\rfloor \end{aligned} \tag{8}$$

According to the *Hash* value, the particle information, including position, density, pressure, etc., is reordered, as shown in Fig. 3. Firstly, the *Hash* value is calculated according to the particle position, then the *sort_by_key* function in the *thrust* library [26] is called to reorder the particles according, and finally, the relevant variables are sorted according to the new particle index.

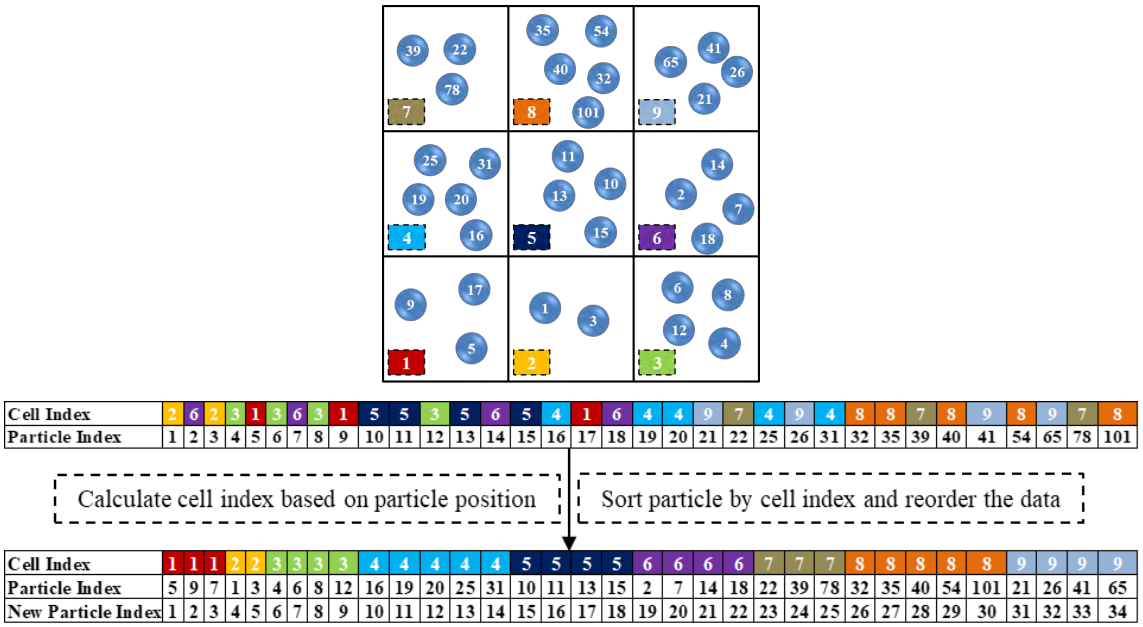


Figure 3. Sort and reorder particle by cell index

Using the reordered particle information for SPH calculation, the flow chart is shown in Fig. 4. The calculation starts, firstly, the initialization is completed at the host, and then the information is passed to the device, and the *kernel function* is called to carry out the simulation:

Step 1: Calculate the particle *Hash* value according to the particle coordinates;
Step 2: Reorder and sort the particle information according to the *Hash* value;
Step 3: Impose boundary conditions;
Step 4: Solve the variables required for the governing equations;
Step 5: Solve the governing equations by particle interactions;
Step 6: Update variables in the prediction step;
Step 7: Restore the particle storage according to the original particle indexes;
Step 8: Repeat steps 1-5;
Step 9: Update variables in the correction step;
Step 10: Repeat step 7;
Step 11: Output the results.

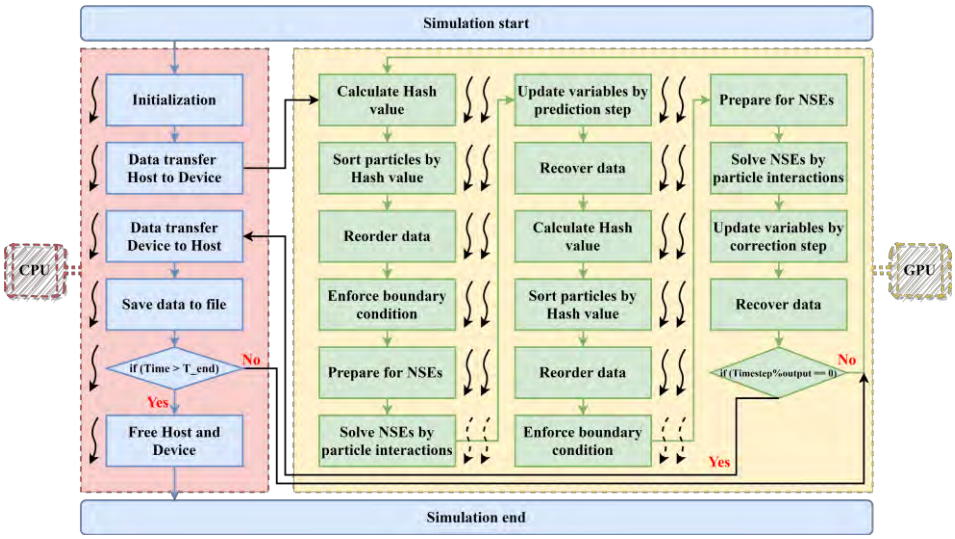


Figure 4. GPU SPH simulation flow chart

3. Numerical validations

3.1 Analysis of speedup

In this section, a 1m×1m 2D hydrostatic water is used for convenience to test the acceleration effect of the developed parallel SPH compared to the conventional serial SPH. The hardware information is shown in Table 1:

Table 1. Hardware information

Item	Configuration
CPU	Intel® Xeon(R) Gold 6230R CPU @ 2.10GHz
RAM	128GB (2394MHz)
GPU	NVIDIA GeForce RTX 3090

The results are shown in Table 2, and the acceleration ratio curve is shown in Fig. 5. When the number of particles is from a few hundred to ten thousand, the GPU computation time is nearly unchanged, the CPU computation time increases more, and the acceleration ratio increases rapidly. From tens of thousands to hundreds of thousands, millions and tens of millions of particles, the GPU computing time also increases, and the acceleration ratio increases in general but the trend gradually moderates. It can be seen that the speedup has reached 169.26 at 16 million particles, and the current parallel SPH model can effectively shorten the computation time and improve the computation efficiency.

Table 2. Result of speedup analysis

Spacing (m)	Particle num	CPU cost (s/K step)	GPU cost (s/K step)	Speedup
0.05	587	130.43	44.96	2.90
0.025	1959	463.12	46.29	10.01
0.0125	7103	1781.37	55.54	32.07
0.01	10881	2800.74	60.07	46.63
0.005	41741	10734.78	88.15	121.79
0.0025	163461	43245.84	283.45	152.57
0.00125	646901	172515.05	1063.58	162.20
0.001	1008615	266718.47	1665.41	160.15
0.0005	4017215	1073433.45	6631.02	161.88
0.00025	16034415	4447248.21	26274.30	169.26

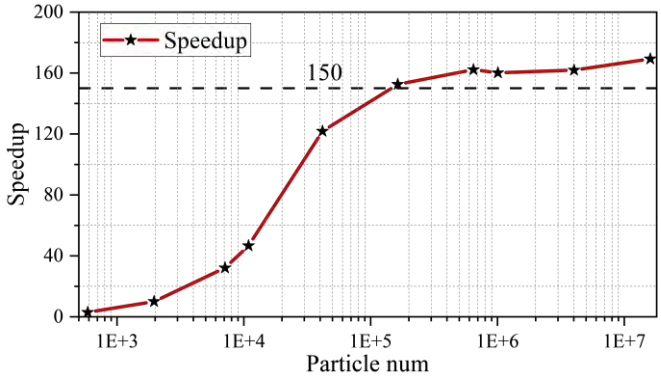


Figure 5. Speedup of present parallel SPH

3.2 Hydrostatic water

In order to verify the stability, accuracy and convergence of the present parallel SPH, a hydrostatic test is taken for analysis in this section. The numerical setup is shown in Fig. 6, with a stationary water column of 1m length and 2m height, and the pressure at the center point of the bottom of the box is monitored. The density of water is 1000 kg/m^3, the artificial

viscosity coefficient is 0.05, and the gravity **g** = -9.81 m/s^2. Three particle spacings, 0.025m, 0.01m and 0.005m are used. The simulation lasts for 10s.

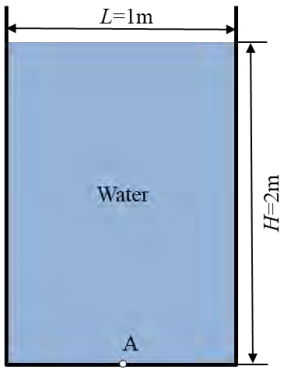


Figure 6. Numerical setup of hydrostatic water

The simulation results are illustrated in Fig. 7. One can see that the pressure contours are stable and accurate without fluctuations. The right panel of Fig. 7 indicate the stability, accuracy and convergence of the present method.

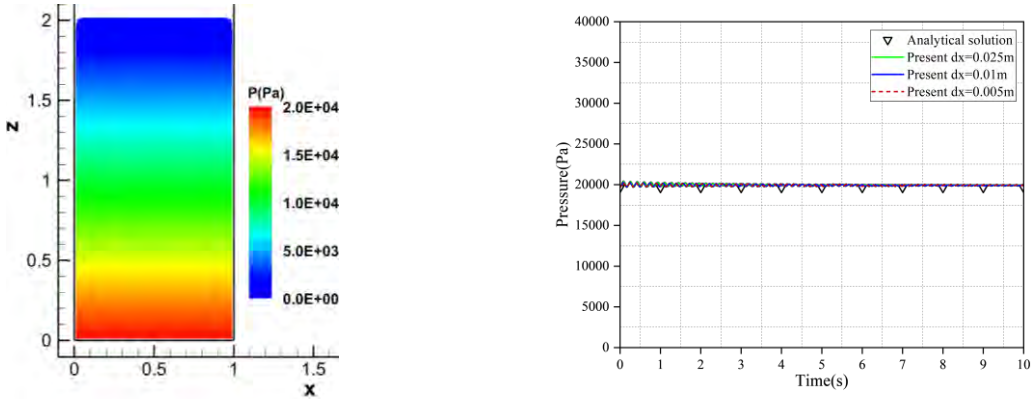
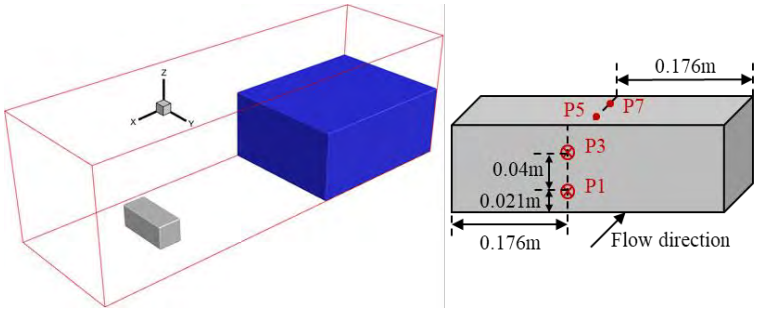


Figure 7. Simulation results of hydrostatic water. Left: pressure contours at t=10s; right: pressure variations at probed point

3.3 3D dam breaking impacting rigid obstacle

In this section, the 3D dam breaking flow is simulated [22]. The numerical setup is shown in Fig. 8. The closed box is 3.22m long, 1m wide and 1m high. To the right of the box, there is a water column 1.228m long, 0.55m high and as wide as the box. On the left side of the box, a rectangular obstacle with a width of 0.403m and a height and length of 0.161m is set. Two measuring points H2 and H4 are placed to monitor the free surface level change at this position. Four pressure probes, P1, P3, P5 and P7, are arranged on the surface of the barrier to record the impacting pressure.



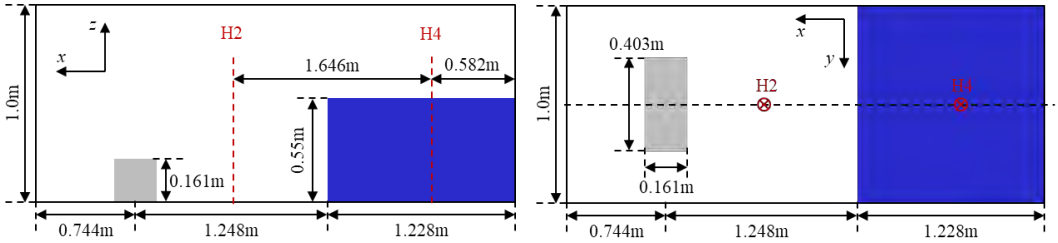


Figure 8. Numerical setup of 3D dam breaking, top: 3D view and pressure sensors on rigid body; second row left: left view; second row right: top view

The simulation results are compared with the experimental snapshots in Fig. 9. At the initial moment, the fluid gradually collapses under gravity and accelerates towards the obstacle. At t=0.4 s, the fluid reaches the front of the obstacle, the middle fluid is blocked and slams turning upward, and the two sides of the fluid continue to move forward and collide with the left wall. The simulation results can accurately reproduce this process and are in good agreement with the experimental snapshots. Fig. 10 shows the simulation results at typical moments, and it can be seen that the fluid will return to the obstacle after hitting the right wall to cause a second impact, which is much weaker than the first one.

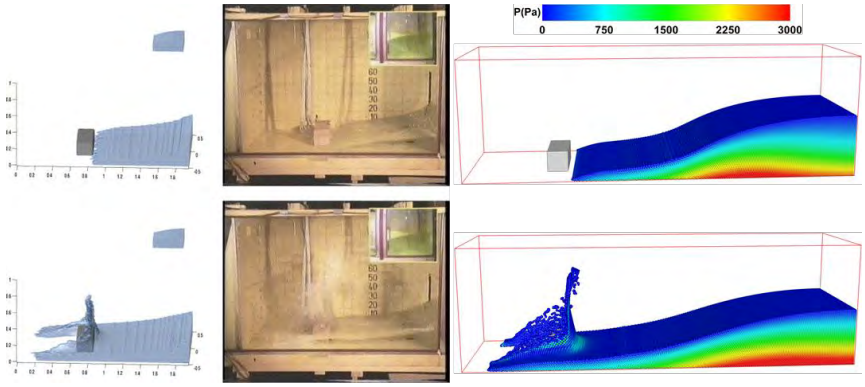


Figure 9. Snapshots of dam breaking from experiment and present method at times t = 0.4 s (top) and 0.56 s (bottom)

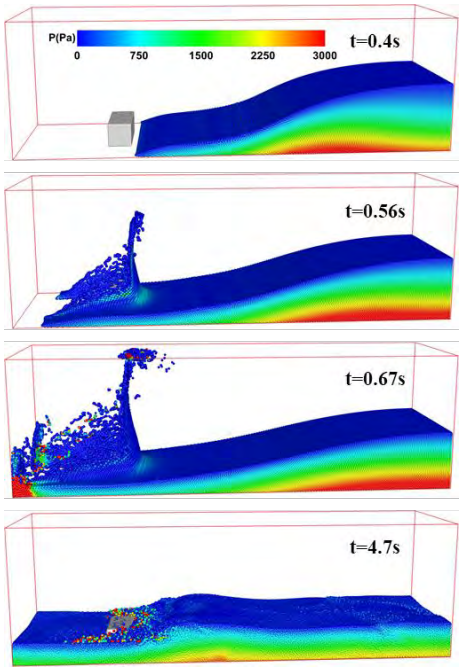


Figure 10. Snapshots of 3D dam breaking at typical time

The free surface evolutions at points H2 and H4 are compared with the reference solutions in Fig. 11. The current method can accurately capture the characteristics of the flow motion in general trend. The ups and downs as well as the peaks of the water level are well predicted. The water height at H4 is more consistent with the experimental data than that of H2. The height variations at H2 have some fluctuations at the moment of 1s-2s, which is related to its location. H2 is in front of the obstacle not far away from it, the impacting fluid moves upward and forms the splash-ups, interfering the measurement of the wave height. H4 is far away from the slamming occurrence and therefore less affected by it.

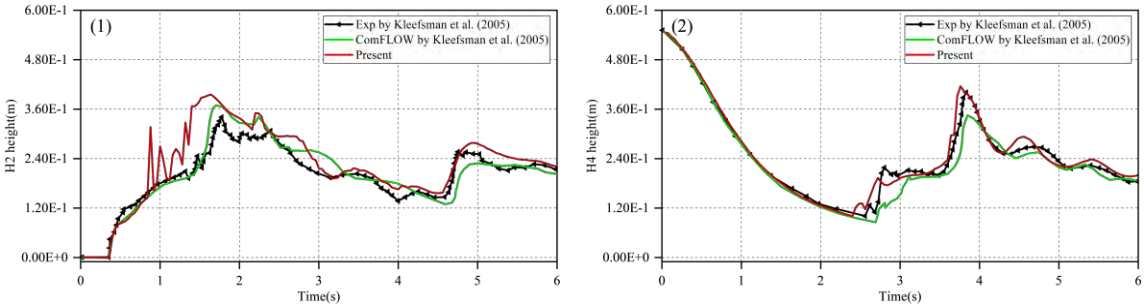


Figure 11. Water height of gauges, left: H2; right: H4

The time histories of the pressure on the obstacle are shown in Fig. 12. All the results can predict the first impacting pressure and the second fluid slamming. The measurements of P1 and P3 are in good agreement with the experimental solution, and there is only an acceptable deviation in the prediction of the pressure peak. The data at P5 and P6 have some pressure noise at the first impacting due to the fact that they are located on the upper surface of the object. The pressure is from the fluid drop after the first slamming, it is reasonable to produce more fluctuations.

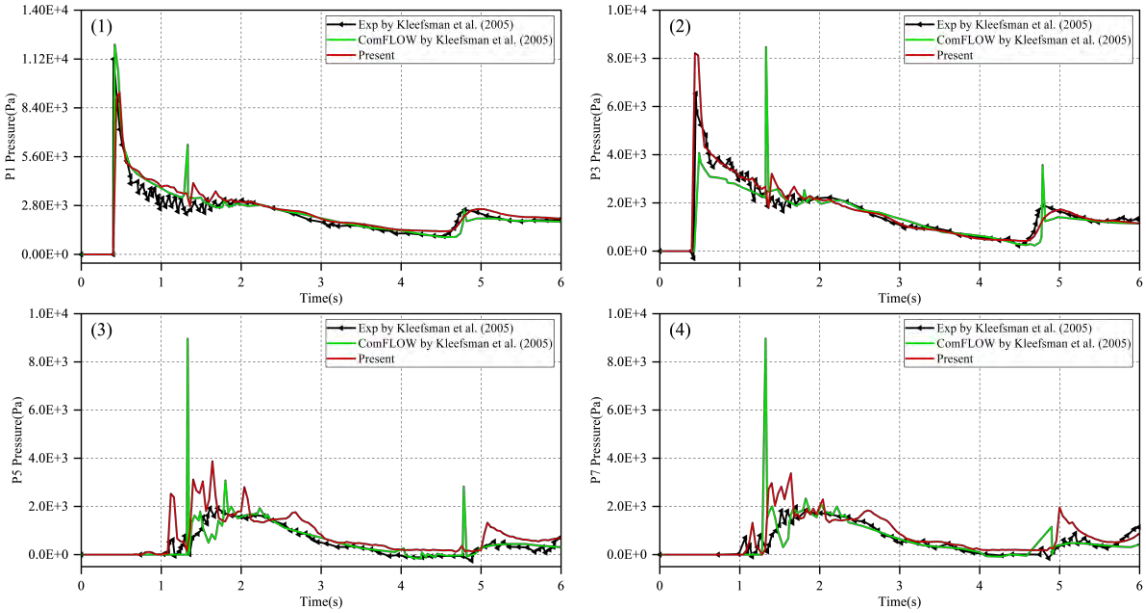


Figure 12. Pressure variations at probed points, (1): P1; (2): P3; (3): P5; (4): P7

4. Conclusion

In this work, an efficient parallel computational model for the SPH method is developed by using CUDA C++ programming. The present model can obtain more than 100 times efficiency improvement compared with the serial code, which is very effective for simulating real engineering problems. The stability, accuracy and convergence are validated by a

hydrostatic water test. The simulation of a 3D dam breaking problem demonstrates the effectiveness of the model in simulating large-scale free surface flow. It not only reproduces the complex flow phenomena, but also accurately predicts the water height and pressure variations, which are in good agreement with the experimental data and reference solutions. In the future, we will consider combining the current model with a solid solver and apply it to the free surface FSI problem to expand its application scenarios.

5. Acknowledgment

This work was partially funded by the National Natural Science Foundation of China (52271307, 52061135107, 52325108), the Liao Ning Excellent Youth Fund Program (2023JH3/10200012), the Liao Ning Revitalization Talents Program (XLYC1908027), and the Dalian Innovation Research Team in Key Areas (No.2020RT03), and the computation support of the Supercomputing Center of Dalian University of Technology.

References

[1] Lucy, L. B. (1977) Numerical Approach to Testing of Fission Hypothesis, *Astronomical Journal* **82**, 1013–1024.
[2] Gingold, R. A. and Monaghan, J. J. (1977) Smoothed Particle Hydrodynamics - Theory and Application to Non-Spherical Stars, *Monthly Notices of The Royal Astronomical Society* **181**, 375–389.
[3] Monaghan, J. J. (1994) Simulating Free-Surface Flows with SPH, *Journal of Computational Physics* **110**, 399-406.
[4] Colagrossi, A. and Landrini, M. (2003) Numerical simulation of interfacial flows by smoothed particle hydrodynamics, *Journal of Computational Physics* **191**, 448–475.
[5] Antuono, M., Colagrossi, A. and Marrone, S. (2012) Numerical diffusive terms in weakly–compressible SPH schemes, *Computer Physics Communications* **183**, 2570–2580.
[6] Sun, P. N., Colagrossi, A., Marrone, S. and Zhang, A.M. (2017) The δ–plus–SPH model: Simple procedures for a further improvement of the SPH scheme, *Computer Methods in Applied Mechanics and Engineering* **315**, 25–49.
[7] Chiron, L., Oger, G., de Leffe, M. and Le Touzé, D. (2018) Analysis and improvements of Adaptive Particle Refinement (APR) through CPU time, accuracy and robustness considerations, *Journal of Computational Physics* **354**, 552–575.
[8] Yang, X., Feng, S., Wu, J., Zhang, G., Liang, G. and Zhang, Z. (2023) Study of the water entry and exit problems by coupling the APR and PST within SPH, *Applied Ocean Research* **139**, 103712.
[9] NVIDIA. (2023) CUDA_C_Programming_Guide_Release12.0.
[10] Harada, T., Koshizuka, S., Kawaguchi, Y. (2007) Smoothed particle hydrodynamics on GPUs. Computer Graphics International. Petropolis: SBC. 63–70.
[11] Zhang, Y. X., Solenthaler, B., Pajarola, R. (2007) GPU accelerated SPH particle simulation and rendering. ACM SIGGRAPH posters.
[12] Crespo, A.J.C., Domínguez, J.M., Rogers, B.D., Gómez-Gesteira, M., Longshaw, S., Canelas, R., Vacondio, R., Barreiro, A. and García-Feal, O. (2015) DualSPHysics: Open-source parallel CFD solver based on Smoothed Particle Hydrodynamics (SPH), *Computer Physics Communications* **187**, 204-216.
[13] Domínguez, J.M., Crespo, A.J.C. and Gómez-Gesteira, M. (2013) Optimization strategies for CPU and GPU implementations of a smoothed particle hydrodynamics method, *Computer Physics Communications* **184**, 617-627.
[14] Zhang, H., Zhang, Z., He, F. and Liu, M. (2022) Numerical investigation on the water entry of a 3D circular cylinder based on a GPU-accelerated SPH method, *European Journal of Mechanics - B/Fluids* **94**, 1-16.
[15] Zhao, Z., Liu, H. and Gong, Z. (2021) A high-efficiency smoothed particle hydrodynamics model with multi-cell linked list and adaptive particle refinement for two-phase flows, *Physics of Fluids* **33**, 064102.
[16] Zhao, Z., Bilotta, G., Yuan, Q., Gong, Z. and Liu, H. (2023) Multi-GPU multi-resolution SPH framework towards massive hydrodynamics simulations and its applications in high-speed water entry, *Journal of Computational Physics* **490**, 112339.
[17] Yang, Q., Xu, F., Yang, Y., Dai, Z. and Wang, J. (2023) A GPU-accelerated adaptive particle refinement for multi-phase flow and fluid-structure coupling SPH, *Ocean Engineering* **279**, 114514.
[18] Zhan, L., Peng, C., Zhang, B. and Wu, W. (2019) A stabilized TL–WC SPH approach with GPU acceleration for three-dimensional fluid–structure interaction, *Journal of Fluids and Structures* **86**, 329-353.
[19] Zhang, Y., Li, J., Xu, Y., Zhu, H. and Liu, Z. (2023) Numerical simulation of dike failure using a GPU-based coupled DEM–SPH model, *Computers & Fluids* **267**, 106090.

[20] Mao, Z., Li, X., Hu, S., Gopalakrishnan, G. and Li, A. (2024) A GPU accelerated mixed-precision Smoothed Particle Hydrodynamics framework with cell-based relative coordinates, *Engineering Analysis with Boundary Elements* **161**, 113-125.
[21] Chen, D., Yao, X., Huang, D. and Huang, W. (2024) A multi-resolution smoothed particle hydrodynamics with multi-GPUs acceleration for three-dimensional fluid-structure interaction problems, *Ocean Engineering* **296**, 117017.
[22] Kleefsman, K.M.T., Fekken, G., Veldman, A.E.P., Iwanowski, B. and Buchner, B. (2005) A Volume-of-Fluid based simulation method for wave impact problems, *Journal of Computational Physics* **206**, 363-393.
[23] Yang, X., Zhang, Z., Zhang, G., Feng, S.and Sun, Z. (2022) Simulating multi-phase sloshing flows with the SPH method, *Applied Ocean Research* **118**, 102989.
[24] Antuono, M., Colagrossi, A.and Marrone, S. (2012) Numerical diffusive terms in weakly-compressible SPH schemes, *Computer Physics Communications* **183**, 2570-2580.
[25] Adami, S., Hu, X.Y.and Adams, N.A. (2012) A generalized wall boundary condition for smoothed particle hydrodynamics, *Journal of Computational Physics* **231**, 7057-7075.
[26] NVIDIA. (2023) CUDA Thrust Release12.0.

A reliability-based design optimization method by instance-based transfer learning and the application of nuclear fuel element design

***Haibo Liu, †Zhe Zhang**

State Key Laboratory of Advanced Design and Manufacturing for Vehicle Body, College of Mechanical and Vehicle Engineering, Hunan University, Changsha,P. R. China, 410082

*Presenting author: liuhaibo619@163.com

†Corresponding author: zhangzhe0828@hnu.edu.cn

Abstract

Reliability-based design optimization (RBDO) is essentially reliability analysis and update process of a series of iteration points.In order to overcome the issues caused by repetitive reliability analysis, this paper proposes an instance-based transfer learning RBDO method. Based on this method, reliability analysis is regard as the target domain, transforming repetitive reliability analysis into a transfer learning problem where the source domain remains invariant while the target domain is continuously updated. By collecting training samples from the source domain, a feedforward neural network(FNN) is first trained for the source domain to establish the mapping relationship between input data and responses in the source domain.To accurately analyze the reliability of the target domain, an instance-based transfer learning method is adopted to reconstruct the FNN for the target domain using knowledge from the source domain. Subsequently, Monte Carlo simulation(MCS) is used to analyze the reliability of the target domain, enabling reliability analysis without the need for additional function evaluations. Finally, the gradient is calculated using the first-order score function and iteration point is updated. The effectiveness of the proposed method is demonstrated through three numerical examples, and the proposed approach is applied to the design of nuclear fuel elements.

Keywords: Reliability-based design optimization, Repetitive reliability analysis, Instance-based transfer learning, Nuclear fuel element design

Engineering application-Nuclear fuel element design

Figure 1. A nuclear fuel element design problem

Nuclear fuel element is the most important component in pressurized water reactor and the core basic component of nuclear industry, serve as the energy source for generating nuclear power and is hailed as the "core" of the nuclear industry. During the actual operation of nuclear reactors, there are numerous uncertainty factors regarding the physical properties of the heat transfer medium and the operating conditions within the reactor. These uncertainties propagate, couple, and amplify within the multi-field system composed of flow fields, temperature fields, etc., leading to significant fluctuations in the thermal-hydraulic characteristics of fuel element within the reactor core, and may even cause safety issues. Reliability-based design optimization (RBDO), considering the impact of various uncertainty factors on the fluctuation of design performance, can achieve a balance between safety and economy in the obtained design solutions. Therefore, it is gradually being applied to the design of nuclear fuel element.

In a nuclear fuel element, the peak surface temperature of the fuel rods is a key parameter that directly impacts the heat transfer performance and safety of the nuclear fuel components. To ensure the safe operation of nuclear fuel element, the thermal design criteria for the core of a nuclear fuel element stipulate that the peak surface temperature of the fuel rods must be below a certain limit to maintain its integrity and prevent the leakage of radioactive materials. However, the peak surface temperature of the fuel rods is not necessarily better the lower it is. It is necessary to comprehensively consider factors such as overall heat transfer efficiency and pressure drop to ensure the safety and economic viability of the reactor during long-term operation.

Table 1. Random variable and parameter distribution information

Random variable and parameter	character	mean	standard deviation	d^l	d^u
Inlet velocity of cooling water	X_1	d_1	0.04	3.2	3.6
Power factor in zone A	X_2	d_2	0.01	0.95	1.05
Power factor in zone C	X_3	d_3	0.01	0.76	0.84
Power factor in zone B	P	$\mu_P=\frac{23.4-d_2-16d_3}{8}$	0.01	/	/

Therefore, this paper aims to minimize the peak surface temperature of the fuel rods with overall heat transfer efficiency and cooling water pressure drop as constraint conditions. Under a constant total heat flux density, a reliable design optimization of the fuel assembly is conducted. The inlet velocity of the cooling water and the power factors of heat flux density in different regions of the fuel rods are treated as random variables, with the mean values of these random variables as design variables. Regarding the distribution of heat flux density in the fuel assembly, the power factors of heat flux density in different radial regions of the fuel rods are primarily considered. As shown in Figure 3, the 5×5 fuel element is divided into three regions, A, B, and C, each containing 1, 8, and 16 fuel rods, respectively. The distribution information of random variables and parameters is shown in the table.

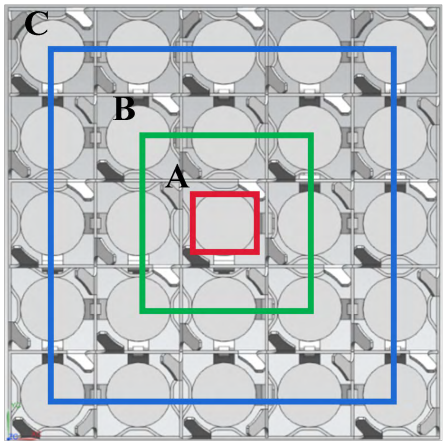


Figure 3. Area division of heat flux

A reliability-based design optimization is required to obtain the obtain a design that balances safety and economy.The RBDO model of nuclear fuel element design is formulated as below:

$$
\begin{aligned}
&\min \ T_{rod\_max}(\mathbf{d}) \\
&\text{s.t.} \ \ Pr\left[G_j(\mathbf{X},P)\geq 0\right]\leq P_{Fj}^t,\ j=1,2 \\
&\quad G_1(\mathbf{X},P)=-\Delta T_{water}+2.0 \\
&\quad G_2(\mathbf{X},P)=\Delta P_{water}-4400 \\
&\quad \mathbf{d}^l=[3.2,0.95,0.76],\mathbf{d}^u=[3.6,1.05,0.84],\mathbf{d}^l\leq\mathbf{d}\leq\mathbf{d}^u \\
&\quad P_{F1}^t=P_{F2}^t=0.005,\mathbf{d}^0=[3.4,1.0,0.8]
\end{aligned}
$$

* MERGEFORMAT (1)

Where, T_{rod_max} represents the maximum surface temperature of the fuel rod; ΔT_{water} indicates the temperature rise of cooling water, which is taken as the heat transfer efficiency constraint; ΔP_{water} represents the cooling water pressure drop. According to the model simulation results and previous experience, the minimum cooling water temperature rise is set as 2°C, and the maximum cooling water pressure drop is set as 4400Pa. $P_{Fj}^t,\ j=1,2$ is the target failure probability.

A finite element analysis (FEA) model, as shown in Figure 2, is established to compute the responses of the nuclear fuel element design problem. Due to the high computational cost of CFD simulations, 80 simulations were used to construct the source domain feed-forward neural network models for the target function and performance functions.

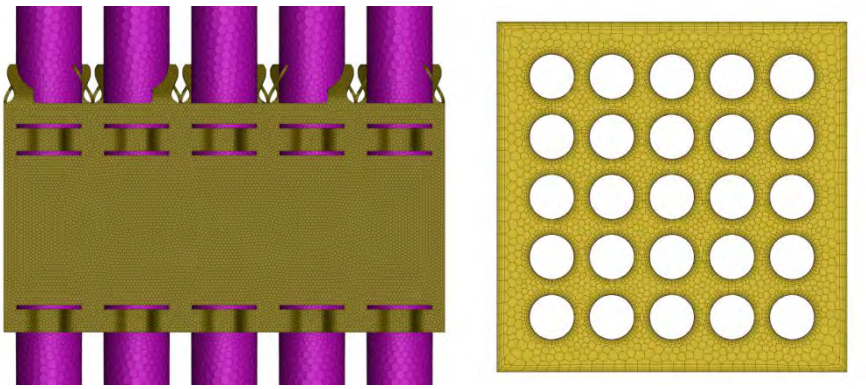


Figure 2. FEA model of nuclear fuel element

Starting from the initial iteration point $\mathbf{d}^0=[3.40,1.00,0.80]$, the proposed method obtains the optimal design $\mathbf{d}^*=[3.289,1.05.0.84]$ after 28 iterations, and the convergence history of the objective function is shown in Figure 4. In each optimization iteration, instance -based transfer learning is used for reliability analysis. Figure 5 (a) and (b) show the change of the failure probability of the two probability constraints during the optimization process, respectively. The optimization results of design variables are shown in Table 2. The results show that, through the reliability design optimization of the inlet velocity of cooling water and

the heat flux of fuel rods in each area, compared with the initial design, the maximum surface temperature of fuel rods is reduced by 2.6°C by the optimal design, and the temperature rise of cooling water is greater than 2°C and the pressure drop is less than 4400Pa with a probability of more than 99%. The safety of fuel assembly is improved and the economic benefit is guaranteed.

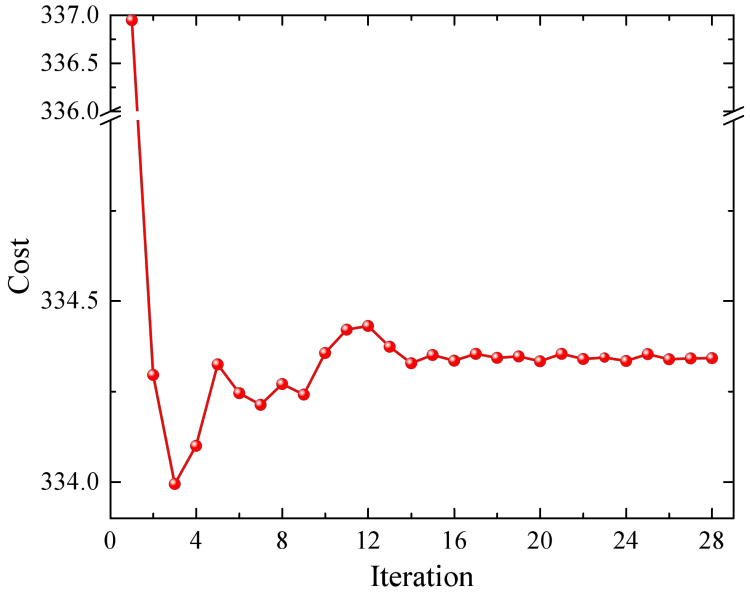


Figure 4. Convergence history of the objective function

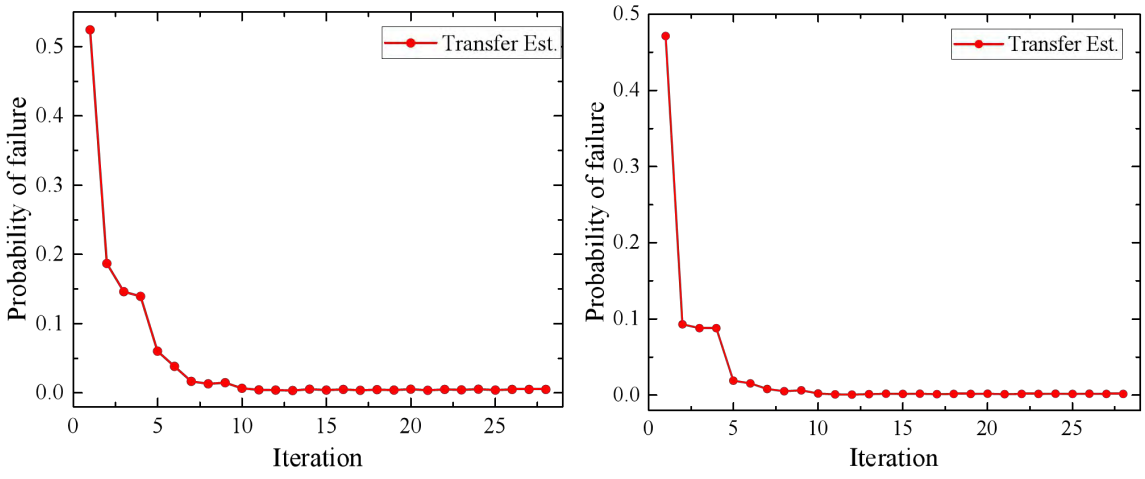


Figure 5. Failure probability of constraints

Table 2. Results of nuclear fuel element design

	d_1	d_2	d_3	Cost	P_{F1}	P_{F2}
Initial point	3.40	1.00	0.80	336.94	0.5242	0.4712
Optimal point	3.289	1.05	0.84	334.34	0.0054	0.0022

References

[1] Navarro M A, Santos A A C. Evaluation of a numeric procedure for flow simulation of a 5×5 PWR rod bundle with a mixing vane spacer. Progress in Nuclear Energy, 2011, 53(8): 1190-1196

[2] Du X, Chen W. Sequential Optimization and Reliability Assessment Method for Efficient Probabilistic Design. Journal of Mechanical Design, 2004, 126(2): 225-233

[3] Liang J, Mourelatos Z P, Nikolaidis E. A Single-Loop Approach for System Reliability-Based Design Optimization. Journal of Mechanical Design, 2007, 129(12): 1215-1224

Estimation of thermal cumulative equivalent minutes for pulse heated tissue

† *Kuo-Chi Liu and Zheng-Jie Chen

Department of Mechanical Engineering, Far East University, Tainan, Taiwan

*Presenting author: kcliu@mail.feu.edu.tw
†Corresponding author: kcliu@mail.feu.edu.tw

Abstract

The temperature changes with time during thermal therapy. In order to ensure the quality and comparative improvement of medical treatment, for a given actual treatment method, it is necessary to determine a certain equivalent thermal dose estimation model. Therefore, validation of procedures for measuring thermal dose in relation to biological effects is clinically required. When a high-intensity heat source is applied to thermal therapy, there is an obvious delay in the heat transfer behavior in biological tissues. In this paper, the dual phase lag bioheat transfer equation was used to analyze the thermal response in the tissue heated by the pulse train, and the CEM43 and the modified CEM43 models were used to estimate the thermal dose during the heating process. The comparison of the CEM43 model with the modified CEM43 model was done for the effect on thermal dose estimation. The results showed that the sensitivity of CEM43 model to temperature is higher for that the temperatures are greater than 55 ^{0}C.

Keywords: Laser-irradiated biological tissue, non-Fourier, thermal dose

Introduction

Biological tissue is a non-uniform heat conductor. In the state of rapid heating, the non-Fourier effect of heat conduction would be highlighted, which is able to affect the thermal reaction in the tissue. The non-Fourier behavior of heat conduction in living biological tissue can be an important factor affecting the quality of treatment. For example, the thermal delay characteristics of biological tissue will delay the time of thermal treatment peak temperature, resulting in lower thermal dose requirements [1]. The finiteness of heat transfer speed will form heat stagnation [2]-[3], which may change the calculation results of equivalent heat flux. It is worth exploring. In order to more accurately predict the heat transfer behavior in biological tissues, D.Z. Tzou [4] proposed a dual-phase lag (DPL) model to further consider the interaction effect of microstructures, and explain the interaction effects of micro-structures with the temperature gradient lag time.

The model CEM43, which is commonly used to evaluate thermal dose and estimate cumulative equivalent minutes at 43 °C, is considered unsuitable for high-temperature rapid heating thermal therapy, and the multiplication coefficient should be corrected from a constant to a temperature function [5]. In order to understand the impact of these discussions on the evaluation of equivalent thermal dose, this paper analyzes the biological heat transfer behavior induced by laser pulse trains based on the dual phase lag mode of bioheat transfer. The corresponding equivalent thermal dose was calculated with the constant and the temperature-dependent multiplication coefficients. The impact of the thermal dose mode and the non-Fourier effect on the estimation of thermal isoeffective dose are presented.

Study method

The bioheat transfer model

In order to solve the paradox occurred in the classical heat flux model and to consider the effect of micro-structural interactions, the DPL model was suggested [4] with

$$\vec{q}(t+\tau_q) = -k\nabla T(t+\tau_T) \tag{1}$$

where T is the temperature, k the heat conductivity, q the heat flux, and t the time. τ_q means the phase lag of the heat flux and τ_T means the phase lag of the temperature gradient.
The general format requirements are as follows:

Eq. (1) is, usually, developed in the first-order Taylor series expansion [4]. Thus, it is rewritten as

$$(1+\tau_q \frac{\partial}{\partial t})\vec{q} = -(1+\tau_T \frac{\partial}{\partial t})k\nabla T \tag{2}$$

Substituting (2) into the energy conservation leads to the DPL equation of bio-heat transfer with constant physiological parameters as the following:

$$\begin{aligned}&(1+\tau_T \frac{\partial}{\partial t})k\nabla^2 T = \\ &(1+\tau_q \frac{\partial}{\partial t})[\rho c \frac{\partial T}{\partial t} - w_b \rho_b c_b (T_b - T) - q_m - q_r]\end{aligned} \tag{3}$$

where t is time. ρ, c, and T denote density, specific heat, and temperature of tissue. c_b and w_b are, respectively, the specific heat and perfusion rate of blood. q_m is the metabolic heat generation and q_r is the heat source for spatial heating. T_b is the arterial temperature.

Estimation models of thermal dose

In the analysis process related to heat therapy, the heating time is often regarded as continuous and long-lasting, and the Arrhenius equation [6] is used to judge the degree of thermal damage in order to control the therapeutic thermal dose. S.A. Sapareto and W.C. Dewey [7] believed that thermal damage is the time cumulative amount of thermal dose, that is, thermal damage is a function of time, so they proposed the following equivalent temperature-time model to calculate equivalent thermal dose.

$$\mathrm{TID(t)} = \int_0^t R_u^{(T_{ref} - T(t'))}\, dt' \tag{4}$$

In the equation, TID is the thermal dose in cumulative equivalent minutes at the reference temperature T_{ref}, dt' is the time at the temperature T (°C), and t represents the final time point of the thermal treatment. The reference temperature is usually set at 43°C, and the calculated heating equivalent time is called cumulative equivalent minutes at 43° C。For temperatures < 43 °C the Ru value is approximately 0.25 and for temperatures > 43 °C the Ru value is approximately 0.5. This thermal dose model is called CEM43 model [5].

Recently, P.X.E. Mouratidis et al. [5] argued that thermal isoeffective doses (TIDs) for predicting biological effects of rapid thermal ablation (eg, using >55 °C) have not been convincingly validated. Therefore, the experimental data obtained at high temperature is used to correct the estimated coefficient Ru of the equivalent thermal dose CEM43 based on the

Arrhenius model as a nonlinear temperature function, as described in equation (5). The modified thermal dose mode is referred to as modified CEM43 model.

$$R_u(\mathrm{t}) = 0.42\times e^{0.0041\times T} \tag{5}$$

Calculation example

This paper considers applying a wide laser beam with uniform irradiance (ϕ_{in}) to biological tissue of thickness L at time t = 0^+.The spot size of the wide laser beam is much larger than the thickness of the heat-affected zone, and then the one dimensional problem is analyzed.

The initial conditions are

$$T(x,0) = T_b \text{ and } \frac{\partial T(x,0)}{\partial t} = \frac{\partial^2 T(x,0)}{\partial t^2} = 0 \tag{6}$$

For UV and IR wavelengths, laser light can be absorbed near the illuminated boundary surface, and laser heating can reasonably be considered as a boundary surface heat flux. This study assumes that the laser irradiation is a pulse train as shown in Figure 1. In Figure 1, ϕ_{in} is the incident laser irradiance, q_0 is the maximum laser intensity, t_p is the pulse width and t_s is the pulse interval.

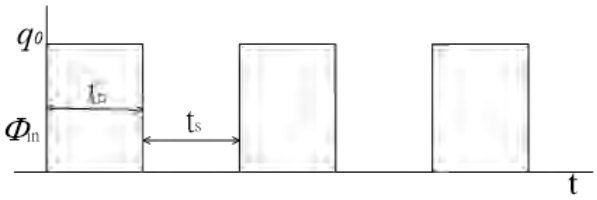


Figure 1. A schematic diagram of a square pulse train.

The boundary conditions are

$$q = \phi_{in}(1\text{-}R_d) \qquad \text{at } x = 0 \text{ for } t > 0 \tag{7}$$

$$T = T_b \qquad \text{at } x = L \text{ for } t > 0 \tag{8}$$

R_d is the diffuse reflectance of the laser light on the illuminated surface.

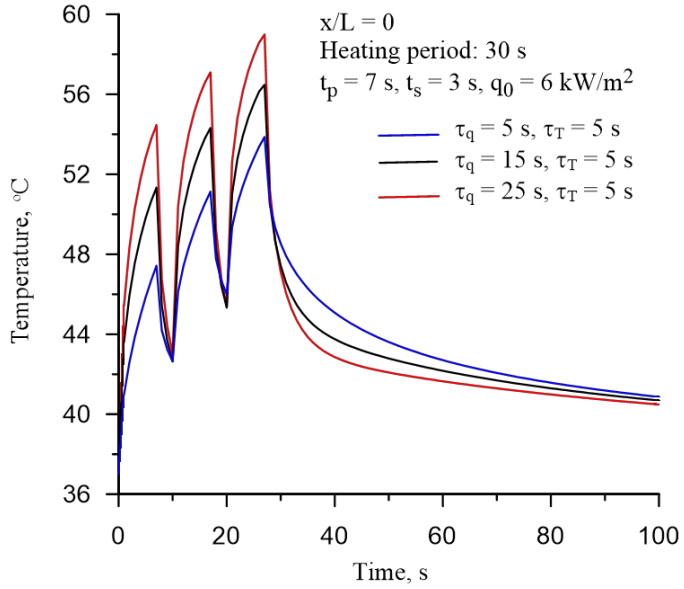
Results and Discussion

This paper refers to the literature [8] for the values of the calculation parameters. These values are q_0 = 20 kW/m^2 , R_d = 0.0528, L = 9 mm, ρ = 1190 kg/m^3 , c = 3600 J/kg·K, ρ_b = 1060 kg/m^3, c_b = 3770 J/kg·K, w_b = 1.87 × 10^{-3} s^{-1}, k = 0.235 W/(m·K). For comparison, the values of some parameters will be adjusted and be stated in the figures.

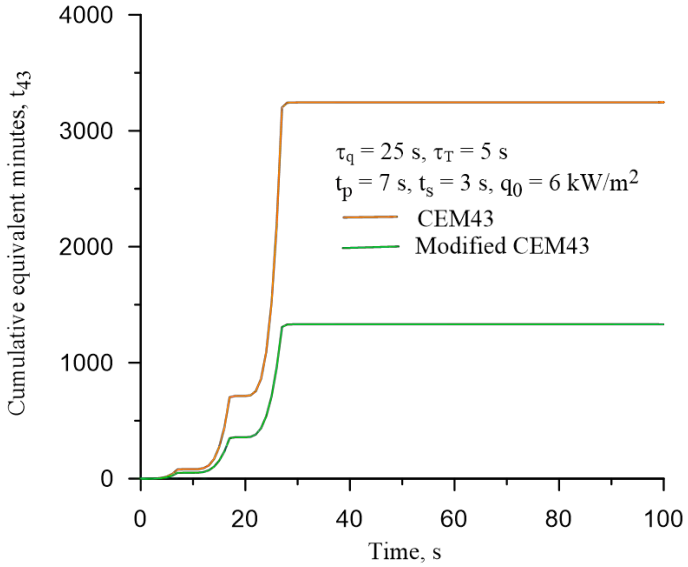
Figure 2 shows the results for x/L = 0 with t_p = 7 s, t_s = 3 s, and q_0 = 6 kW/m^2. It includes three parts: (a) temperature variations based on various sets of the values of τ_q and τ_T, (b) the corresponding cumulative equivalent minutes for the set of τ_q = 25 s and τ_T = 5 s, and (c) the corresponding cumulative equivalent minutes for the sets, τ_q = 15 s and τ_T = 5 s and τ_q = 5 s and τ_T = 5 s. As shown in Figure 1(a), the temperature of the end face x = 0 increases gradually with each laser pulse heating. The heating stops during the pulse interval so that the temperature drops until the next laser pulse heating starts and the temperature rises again. It is known that the phase lag of heat flux, τ_q, can reduce the velocity of heat transfer so that heat would accumulates on the end face. As a result, the temperature at the end x = 0 increases as

the value of τ_q increases. For $\tau_q = \tau_T$, the dual phase lag bioheat transfer equation can reduce to the Pennes bioheat transfer equation. The behavior of heat conduction in the tissue will follow Fourier law for $\tau_q = \tau_T = 5$ s.

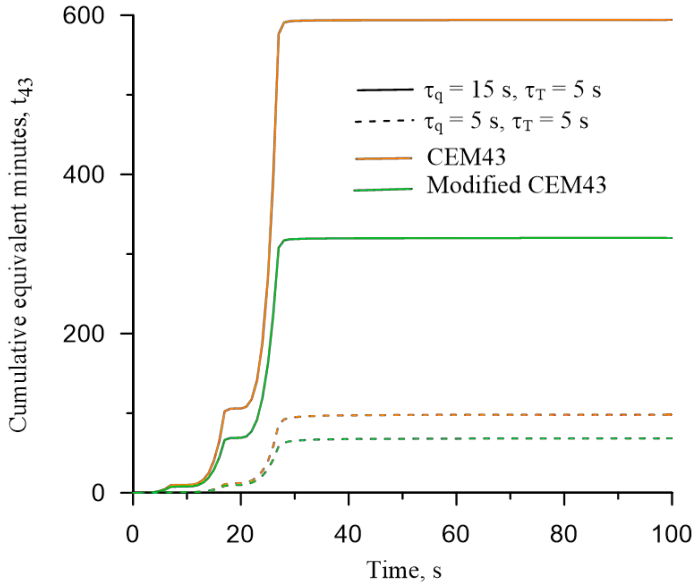
Figure 2(b) presents the cumulative equivalent minutes evaluated by the CEM43 and the modified CEM43 models for the case of $\tau_q = 25$ s and $\tau_T = 5$ s. It is observed that the evaluation values obtained by the CEM43 and the modified CEM43 models are almost the same for that the temperature is less than 55 ^{0}C. After the temperature reaches 55 ^{0}C, the evaluation values of cumulative equivalent minutes dramatically rise for the CEM43 and the modified CEM43 models. As observed, there is an obvious difference between the variation curves of cumulative equivalent minutes based on the CEM43 and the modified CEM43 models. The CEM43 model is more sensitive to the temperature that is greater than 55 ^{0}C.



(a)



(b)



(c)

Figure 2. The results for x/L = 0 with t_p = 7 s, t_s = 3 s, and q_0 = 6 kW/m^2: (a) temperature variations based on various sets of the values of τ_q and τ_T, (b) the corresponding cumulative equivalent minutes for the set of τ_q = 25 s and τ_T = 5 s, and (c) the corresponding cumulative equivalent minutes for the sets, τ_q = 15 s and τ_T = 5 s and τ_q = 5 s and τ_T = 5 s.

The comparison of the CEM43 model with the modified CEM43 model is still made in Figure 2(c). The cumulative equivalent minutes evaluated by the CEM43 and the modified CEM43 models for the cases, τ_q = 5 s and τ_T = 5 s and τ_q = 15 s and τ_T = 5 s, are displayed in Figure 2(c). The sensitivity of the CEM43 model to the higher temperatures is also able to be found. As shown in Figure 2(a), the temperatures are below 55 ^{0}C for τ_q = 5 s and τ_T = 5 s so that the evaluation value of the cumulative equivalent minutes is below 100 minutes, not over the threshold value. For the case of τ_q = 15 s and τ_T = 5 s, the values of t_{43} based on the CEM43 and the modified CEM43 models are over the threshold value, 240 minutes [3], for hyperthermia until the third pulse heats.

Conclusions

This paper analyzes the dual phase lag bioheat transfer induced by laser pulse trains. The corresponding equivalent thermal doses were estimated based on the CEM43 model and the modified CEM43 model, respectively. The predicted temperature is higher with the larger value of the phase lag of heat flux, τ_q, because τ_q makes the velocity of heat transfer decreasing so that heat accumulation is created on the end face. And then, the evaluation value of the cumulative equivalent minutes will be higher. In other words, the non-Fourier effect is actual on the estimation of thermal isoeffective dose. The results illustrate that the evaluation values obtained by the CEM43 and the modified CEM43 models are almost the same for below 55 ^{0}C. Over 55 ^{0}C, the evaluation values of cumulative equivalent minutes become sensitive to temperature for the CEM43 and the modified CEM43 models.

References

[1] Shih, T.C., Kou, H.S., Liauh, C.T., Lin, W.L. (2005) The impact of thermal wave characteristics on thermal dose distribution during thermal therapy: a numerical study, *Medical Physics* **32**, 3029 – 3036.
[2] Rastegar, J.S. (1989) Hyperbolic heat conduction in pulsed laser irradiation of tissue, in M. J. Berry and G. M. Harpole (eds.), Thermal and Optical Interactions with Biological and Related Composite Materials: Proc. SPIE, Los Angeles, CA, vol. 1064, pp. 114–117, SPIE Press, Bellingham, WA.
[3] Kaminski, W. (1990) Hyperbolic heat conduction equation for materials with a nonhomogeneous inner structure, *ASME Journal of Heat and Mass Transfer* **112**, 555–560.
[4] Tzou, D.Y. (1996) Macro- to Microscale Heat Transfer: the Lagging Behavior, Taylor & Francis, Washington, DC.
[5] Mouratidis, P.X.E., Rivens, I., Civale, Symonds-Tayler, J., and Haar, R., G. (2019) Relationship between thermal dose and cell death for "rapid" ablative and "slow" hyperthermic heating, *International Journal of Hyperthermia* **36**, 228–242.
[6] Dewey, W.C. (1994) Arrhenius relationships from the molecule and cell to the clinic. *International Journal of Hyperthermia* **10**, 457–483.
[7] Sapareto, S.A. and Dewey, W.C. (1984) Thermal dose determination in cancer therapy. International Journal of Radiation Oncology - Biology - Physics **10**, 787–800.
[8] Liu, K. C. and Leu, J.S. (2023) Heat transfer analysis for tissue with surface heat flux based on the non-linearized form of the three-phase-lag model. *Journal of Thermal Biology* **112**, 103436.

Lamellar Vortex Generator around Submarine Hull Sail: ANN/GA-based Sound Source/Structural Optimization

Xinxin Meng[1*], Yangjun Wang[2], Yuan Zhuang[1] , Decheng Wan[1†]

[1] Computational Marine Hydrodynamics Lab (CMHL), School of Naval Architecture, Ocean Engineering, Shanghai Jiao Tong University, Shanghai, China

[2] College of Advanced Interdisciplinary Studies, National University of Defense Technology, Nanjing, China

*Presenting author: mengxinxin@sjtu.edu.cn

†Corresponding author: dcwan@sjtu.edu.cn

Abstract

Stealth performance is directly related to the probability of submarines being detected underwater, with self-generated flow noise being one of the key factors. Flow noise can be roughly divided into monopole, dipole, and quadrupole sound sources, among which the contribution of dipole sound sources is the most significant during low-speed navigation. Vortex generators significantly suppress local dipole sound sources. This paper aims to reduce dipole sound sources intensity by optimizing layered vortex generators, and then enhance submarine stealth performance. This paper examines the influence of lamellar vortex generator height (h), angle (θ), and position coordinates (x, y) on dipole sources based on the r roadband ooise Source ood el (r oSo). Sampling is conducted using the Response Surface oe thodology (RSo) and Optimal Latin Hypercube Sampling (OLHS), and over 50 datasets are obtained through Computational Fluid Dynamics (CFD) simulations. Artificial oe ural oe tworks (Ao o) combined with genetic algorithm (GA) is used to employ for parameter optimization of the lamellar vortex generator, and the optimized result is compared with the regression result from the traditional model, and validations are performed using CFD simulations. Finally, the paper delves into the impact of vortex generators on submarine sound sources through comparative analysis of the flow and sound field distributions between the optimized and original models.

Keywords: dipole sound source, vortex generator, few-shot learning, artificial neural network

Introduction

The sail of underwater vehicles generates various vortex structures, including horseshoe vortices, gamma vortices, and necklace vortices. These vortices contribute to non-uniform axial velocity in the propeller disk and increased underwater flow noise. It has been confirmed that installing vortex generators around the sail can effectively improve the axial velocity uniformity of the propeller disk [1]. Liu et al. (2014) explored the mechanism by which vortex generators improve the axial velocity uniformity of the propeller disk [2], finding that vortex generators disrupt the horseshoe

vortices around the sail, thereby enhancing velocity uniformity. With advancements in computational performance, more detailed investigations have been conducted using refined Large Eddy Simulation (LES) turbulence models [3]. Researchers discovered that vortex generators produce small vortices opposite in direction to the horseshoe vortices, weakening the horseshoe vortices until they completely dissipate at the submarine's stern. Due to their small size, vortex generators do not significantly alter drag and lift. However, they modify the near-wall flow, affecting the fluctuating pressure on the wall. Consequently, the presence of vortex generators inevitably alters the flow noise characteristics of underwater vehicles. Liu et al. compared the impact of vortex generators with different shapes, positions, and angles of attack on flow noise using a simplified model [4]. They found that vortex generators have a significant noise reduction effect. However, they directly provided the shapes and positions of the vortex generators without explaining the underlying rationale.

oa chine learning methods have long been popular tools due to their powerful capabilities for arbitrary high-order nonlinear fitting, making them favored by researchers. Artificial o eural oe tworks (Ao o) are one such method, which classify and regress problems using automatic differentiation. Thurman et al. explored the mechanism by which different inflow conditions affect the force coefficients of underwater vehicles using artificial intelligence methods [5]. Rostamzadeh-Renani et al. investigated the impact of tail sail vortex generators on the lift and drag of underwater vehicles using artificial intelligence methods, and combined these with genetic algorithms to find the optimal vortex generator parameters [6]. Saghi et al. (2022) used Ao o to regress the relationship between the length-to-diameter ratio and Reynolds number with the drag coefficient of underwater vehicles [7]. While Aoo have been extensively integrated with computational fluid dynamics, simple equation fitting can be more satisfactory when the training samples are limited. r esides increasing the number of training samples to improve the fitting accuracy of Ao o , small-sample training is a suitable approach. Zhang et al. explored the relationship between airfoil shape and lift-to-drag ratio using a meta-learning approach [8]. The results demonstrated that a high-accuracy surrogate model could be obtained with only a small number of computational samples. Leng et al. combined the concept of small-sample training by sequentially inputting data of different computational accuracies into the neural network [9], thereby enhancing training accuracy while reducing the number of high-precision training samples required.

The mainstream approach of calculating underwater flow noise is the hybrid method, which involves solving the acoustic field through CFD flow simulations using the FW-H method or Kirchhoff method. However, detailed simulations become impractical when the number of cases is large. Therefore, this paper considers an approximate method—the broadband noise source model, which will be described in detail in Section 2.2. Since this model relies on RAo S equations, it reduces computational effort at the cost of some accuracy. Consequently, more detailed and in-depth studies will be conducted in future work.

To accurately capture the relationship between design variables and target variables and

overcome limitations in the size of the training dataset, this study employs few-shot training approach. Initially, a certain number of sample cases are obtained through response surface methodology, and a relatively rough surrogate model is developed via regression using Design Expert. Subsequently, a large training dataset is generated based on the rough surrogate model. This dataset is utilized for pre-training an Ao o model. Following this, a new set of sample cases generated using Optimal Latin Hypercube Sampling (OLHS) is used for training. The Ao o architecture remains consistent with the pre-training, with the initial parameters derived from the pre-training phase. This process yields the final trained model. The conceptual framework is illustrated in Fig. 1.

This study presents several innovations:

- Investigating the acoustic characteristics of the sail and surrounding solid walls influenced by the vortex generator based on a broadband noise source model.
- Improving the regression accuracy of abstract problems by employing a small-sample training approach.
- Utilizing a combination of Ao o and Genetic Algorithm (GA) methods to determine the optimal settings for vortex generator parameters, and exploring the mechanisms by which the vortex generator enhances the distribution of sound sources.

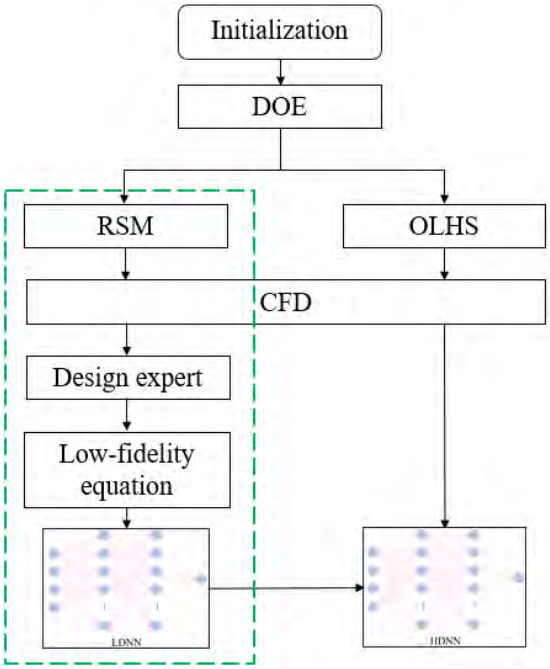


Figure 1. The optimization process

Numerical equations

Numerical Equations

All numerical simulations were conducted using STAR-CCo+ . The flow field calculations were performed using the Reynolds-Averaged oa vier-Stokes (RAo S)

approach. In the RAo S method, the incompressible Reynolds-averaged o avier-Stokes equations were employed, including the continuity equation and the momentum equations:

$$\frac{\partial \overline{u_i}}{\partial x_i} = 0 \tag{1}$$

$$\rho \frac{\partial \overline{u_i}}{\partial t} + \rho \overline{u_j} \frac{\partial \overline{u_i}}{\partial x_j} = \overline{f_i} - \frac{\partial p_i}{\partial x_i} + \frac{\partial}{\partial x_j}\left(\mu \frac{\partial \overline{u_i}}{\partial x_j} - \rho \overline{u_i' u_j'} \right) \tag{2}$$

In these equations, ρ represents density, t denotes time, f stands for body force, p is pressure, and μ is the dynamic viscosity coefficient. The variables x_i indicate the three coordinate directions, and $\overline{u_i'}$ and $\overline{u_j'}$ are the time-averaged velocity fluctuations. Compared to the original oa vier-Stokes equations, the Reynolds-averaged equations include additional Reynolds stress terms $\tau_{ij} = -\rho \overline{u_i' u_j'}$. There are nine components in τ_{ij}, but only six are independent due to symmetry. The original set of equations contains four unknowns, resulting in ten unknowns in total, which prevents the system from being closed. Therefore, a new turbulence model needs to be introduced.

The turbulence model adopted is the SST k-omega model, with the following equations:

$$\frac{\partial k}{\partial t} + \overline{u_j} \frac{\partial k}{\partial x_j} = \frac{\partial}{\partial x_j}\left[(v + \sigma_k v_t) \frac{\partial k}{\partial x_j} \right] + P_k - \beta^* k\omega \tag{3}$$

$$\frac{\partial \omega}{\partial t} + \overline{u_j} \frac{\partial \omega}{\partial x_j} = \&\frac{\partial}{\partial x_j}\left[(v + \sigma_\omega v_t) \frac{\partial \omega}{\partial x_j} \right] + \alpha S^2 - \beta \omega^2 + 2(1 - F_1)\sigma_{\omega 2} \frac{1}{\omega} \frac{\partial k}{\partial x_j} \frac{\partial \omega}{\partial x_j} \tag{4}$$

Turbulent viscosity coefficient

$$v_t = \frac{a_1 k}{max(a_1 \omega, \Omega F_2)} \tag{5}$$

where

$$\sigma_k = \frac{1}{F_1/\sigma_{k,1} + (1 - F_1)/\sigma_{k,2}} \tag{6}$$

$$\sigma_\omega = \frac{1}{F_1/\sigma_{\omega,1} + (1 - F_1)/\sigma_{\omega,2}} \tag{7}$$

The turbulent kinetic energy k is defined as $k = \frac{1}{2} u_i' u_j'$, and the specific dissipation rate ω is given by $\omega = \frac{\varepsilon}{k\beta^*}$ where $\varepsilon = \nu \frac{\partial u_i'}{\partial x_k} \frac{\partial u_i'}{\partial x_k}$. In these equations, β^*, γ, β, and $\sigma_{\omega 2}$ are model constants, and F_1 and F_2 are blending functions.

Broadband noise source model

Using the RAo S turbulence model for continuous computation, the Curle noise source model (Curle, 1955) effectively predicts dipole noise sources. This method is computationally efficient, allowing for rapid prediction of the location and intensity of significant noise sources. According to Curle's equation, the acoustic density can be expressed as:

$$\rho(x,t)' = \frac{1}{(4\pi c_0^3)} \int_S \left[\frac{(x-y)}{r^2} \frac{\partial p}{\partial t}\left(y, t - \frac{r}{c_0}\right) \right] \cdot n \mathrm{d}S(y) \tag{8}$$

where, $\rho(x,t)'$denotes the acoustic density, and c_0 represents the sound speed in the quiescent medium. The term $t - \frac{r}{c_0}$ indicates the emission time. Additionally, p is the surface pressure, y means surface centroid position on solid boundary grid, x describes the position in the far-field, r is the physical distance between the source and far-field positions, and n denotes the outward wall-normal direction. Considering the isentropic for the perturbation, the variation of the sound pressure and density perturbations is given as:

$$p' = c_0^2 \rho' \tag{9}$$

The acoustic density equation (Eq.(8)) can be converted to the following acoustic pressure equation:

$$p(x,t)' = \frac{1}{(4\pi c_0)} \int_S \left[\frac{(x-y)}{r^2} \frac{\partial p}{\partial t}\left(y, t - \frac{r}{c_0}\right) \right] \cdot n \mathrm{d}S(y) \tag{10}$$

In the far field, the acoustic directional intensity per unit area of a solid surface along the x - y is approximated as:

$$\overline{p'^2} = \frac{1}{16\pi^2 c_0^2} \int_S \frac{(\cos\ \theta)^2}{r^2} \overline{\left[\frac{\partial p}{\partial t}\left(y, t - \frac{r}{c_0}\right) \right]^2}\ A_c(y) \mathrm{d}S(y) \tag{11}$$

where, $A_c(y)$, is the acoustic correlation region, and θ represents the angle between r and the wall normal n. The local contribution per unit of Surface Acoustic Power (SAP) can be calculated according to the following equation:

$$SAP = \frac{1}{\rho_0 c_0} \left[\int_0^{2\pi} \int_0^{\pi} p'^{\ 2}\ r^2 \sin\ \theta \mathrm{d}\theta \mathrm{d}r \right] \tag{12}$$

Then the SAP can be calculated from:

$$SAP = \frac{1}{12\pi \rho_0 c_0^3} \int_S \overline{\left(\frac{\partial p}{\partial t} \right)^2}\ A_c(y) \mathrm{d}S(y) \tag{13}$$

where, ρ_0 represents the far-field density, and $\overline{\left(\frac{\partial p}{\partial t}\right)^2}$ is the mean-square time derivative of the surface pressure. The integral is taken over the surface S of the solid boundaries. The formulation can be rewritten to include the acoustic correlation region and $\overline{\left(\frac{\partial p}{\partial t}\right)^2}$ as:

$$SAP(W/m^2) = \frac{1}{12\pi\rho_0 c_0^3} \cdot 4\pi^3 \overline{\left(u'_{\text{turb}}\right)^2} (p'_{\text{wall}})^2 \tag{14}$$

where, p'_{wall} and u'_{turb} represent the wall pressure fluctuation and the mean square turbulent velocity. According to Hinze (1975), these are defined as follows:

$$\begin{aligned} p'_{\text{wall}} &= max\left(3\tau_{\text{wall}}, 0.7\rho_0 \cdot \frac{2}{3}k\right) \\ \overline{\left(u'_{\text{turb}}\right)^2} &= \frac{2}{3} max\left(k, \frac{3.3\tau_{\text{wall}}}{\rho_0}\right) \end{aligned} \tag{15}$$

where, k and τ_{wall} mean the turbulent kinetic energy and the wall shear stress, which can be computed by the RAo S turbulence model to obtain. In this paper, the Curle noise source model is used to calculate the SAP of the SUr OFF submarine, employing the surface of submarine as the integration surface. Convert SAP to the form of decibels and P_{ref} is the reference sound power:

$$SAP(\text{dB}) = 10\log \frac{SAP}{P_{\text{ref}}} \tag{16}$$

In this study, the target variable used is the mean sound power. The values obtained from CFD calculations are the center values of the grid faces, and the areas of different grid faces vary. Therefore, the sound acoustic power (SAP) for each face is first calculated using (Eq. (14)). Subsequently, the area-averaged acoustic power for the region is computed using (Eq. (17)), and finally, it is converted to decibels using (Eq. (18)).

$$\overline{SAP}(\text{W/m}^2) = \frac{1}{\sum_{i=1}^{n} S_i} \sum_{i=1}^{n} SAP_i \times S_i \tag{17}$$

$$\overline{SAP}(\text{dB}) = 10\log \frac{\overline{SAP}}{P_{\text{ref}}} \tag{18}$$

where, i denotes the i-th grid cell, S_i represents the area of the grid cell numbered i, and SAP_i is the acoustic power calculated for the i-th grid cell.

Methodology

Artificial neural network

A typical Ao o model consists of input layer, hidden layer and output layer [10]. Each layer contains a varying number of neurons, and signals from the upper layer are processed by neurons and then transmitted to the next layer by multiplying the corresponding weights. Self-learning is the process of minimizing the error between prediction data and experimental data by constantly updating the weights. The neuron is the basic unit of Ao o, which can be written as

$$y = f\left(\sum_{i=1}^{k} p_i w_i + b\right) \tag{21}$$

The data is preprocessed using normalization, and the activation function $tanh$ is employed. The backpropagation learning algorithm is utilized to minimize the loss

function, updating the weights and biases within the network. The loss function, defined as the mean squared error (oSE) between the calculated and predicted values, is represented by Eq. (22). The network architecture is illustrated in Fig. 2.

$$MSE = \frac{1}{n}\sum_{i=1}^{N}(y_i - y_c)^2 \tag{22}$$

where y_c is the calculated value, y_i is the value predicted by Ao o, a nd o is t he number of training samples.

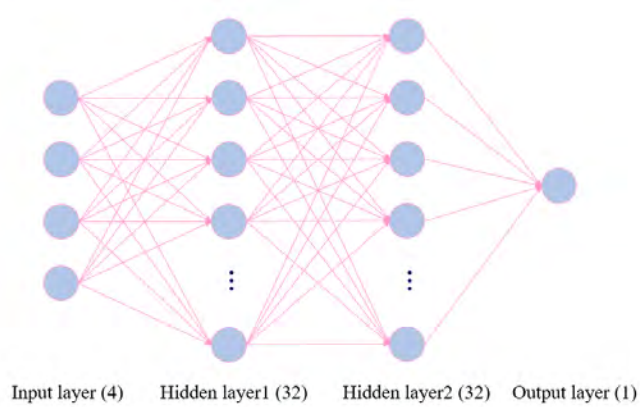


Fig. 2. Composition of Ao o

To describe the accuracy and robustness of the model more intuitively, the correlation coefficient (R2) and mean absolute error (oA E) for model were calculated. The R^2 was often used in regressing problems to reflect how well the predicted values match the experimental values, the value of R^2 closer to 1 indicates a better fit of the model. oAE is the absolute error between the predicted and calculated values, which helps us determine whether the model error is within an acceptable range. The lower oAE refers to the more accurate the model. The R^2 and oAE can be calculated by Eq. (23) and Eq. (24), respectively.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - y_c| \tag{23}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - y_c)^2}{\sum_{i=1}^{n}(\bar{y} - y_c)^2} \tag{24}$$

Where o is the number of training samples, y_i and y_c refer to the predicted value and the calculated value of the i-th sample, $\bar{y}$ represents the mean value of the calculated value.

Response surface method

Response Surface oe thodology (RSo) is a process optimization technique that integrates statistics, mathematics, and computer technology. Compared to orthogonal tests, RSo accurately obtains regression equations between multiple factors and response values [11], and facilitates the analysis of interactions. Through experimental design and result analysis, RSo can identify the relationships between factors and response variables, and determine optimal conditions. When applying RSo to design experiments, research indices, factors, and levels should be determined according to the

research objectives. Design variables were selected based on prior knowledge, and the number of experimental designs was determined according to Eq.(19).

$$H = 2^h + 2h + c \tag{19}$$

where, h represents the number of design variables, $c = 1$, and H denotes the number of experiments.

As a second-order model, RSo is a robust and flexible technique for accurately estimating the relationship between design variables and the response surface [12]:

$$y = \beta_0 + \sum_{i=1}^{k} \beta_i x_i + \sum_{i=1}^{k} \beta_{ij} x_i x_j + \sum_{i=1}^{k} \beta_{ii} x_i^2 \tag{20}$$

where x_i are the design variables, y is the objective function, and β is unknown coefficient.

Optimal Latin hypercube sampling method

The principle of Latin hypercube sampling (LHS) is that if sampling in a multidimensional space, each dimensional space is divided into $q \times q$ subspaces uniformly, and q points are selected randomly to ensure that each point is sampled only once in each subspace [13]. r ased on LHS sampling, Optimal Latin hypercube sampling (OLHS) is obtained by optimizing the sample through the maximum minimum distance criterion [14]. Taking two-dimensional spatial sampling as an example, the comparison between OLHS and LHS is shown in Fig. 3. Although LHS ensures that only one sample is selected at each level, there is a phenomenon of sample points aggregation. The OLHS has better spatial homogeneity, which helps to obtain the main features of the space and improve the initial surrogate model accuracy.

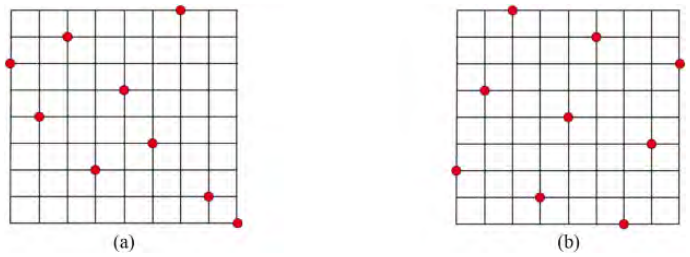
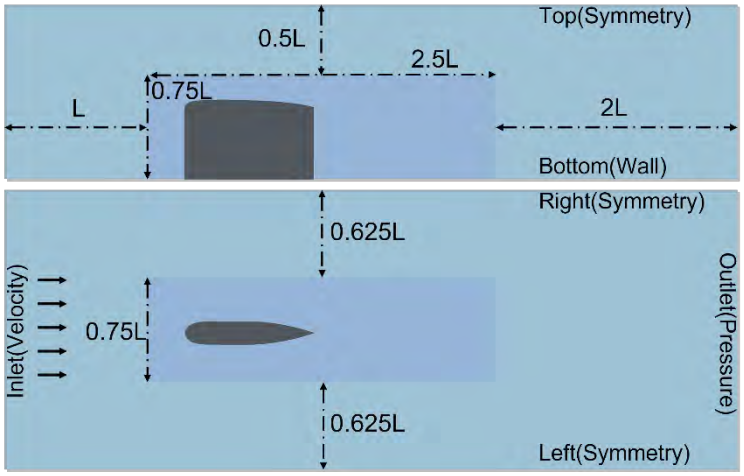


Figure 3. Sample points distribution in two-dimensional space for (a) LHS and (b) Opt-LHS [15].

Grid independence validation

This study performs calculations based on a simplified model, using a sail length L = 0.37 m, and adopts dimensionless representation for the computational domain, as shown in Fig. 4(a), where all relevant information is marked. The dark blue area indicates the mesh refinement region. Traditional Prism Layer o esher are inadequate for generating boundary layers and capturing the horseshoe vortex at the sail-bottom interface. Therefore, the boundary layer mesh is generated using the Advancing Layer

oe sher method in STAR-CCo+ , as depicted in Fig. 4(b). The mesh around the sail is refined, as shown in Fig. 4(c).



(a) domain information

(b) rounda ry layer grid

(c) Sail perimeter grid

Figure 4. Computational domain and local mesh

The background mesh size is set to be 0.5 m. The mesh comprises 26 prismatic layers with an expansion rate of 1.1, and the total thickness of these prismatic layers is 7.5% of the base size. The wall mesh size is defined as 1% of the base size, while the local refinement area is 0.8% of the base size. The volumetric mesh growth rate is specified as 1.05.

To verify grid independence, coarse and fine grids were generated using a base size of 0.5 m as a reference, with $\sqrt{2}$ as a scaling factor. The number of elements in the three sets of grids is provided in Tab. 1. Time discretization is performed using a first-order scheme. The pressure is solved using the Preconditioned Conjugate Gradient (PCG) solver, while the pressure-velocity coupling is addressed with the Semi-Implicit oe thod for Pressure-Linked Equations (SIo PLE). The water density is 1000kg/m^3, and the kinematic viscosity is 1×10^{-6}m^2/s. The time step is set to 1×10^{-3} seconds, with a total simulation time of 2 seconds. The method for $\overline{SAP}$ has been introduced in Sec. (1.2). The calculation results are presented in Tab. 1. The entire computational domain is too large relative to the sail. Therefore, a local target variable is tested within the selected region, as shown in Fig. 5.

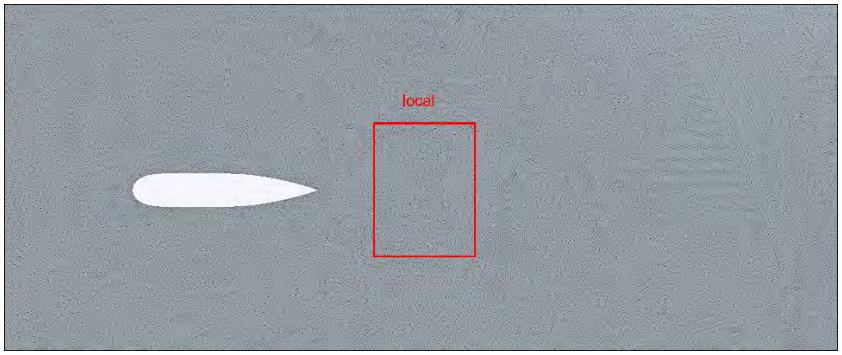
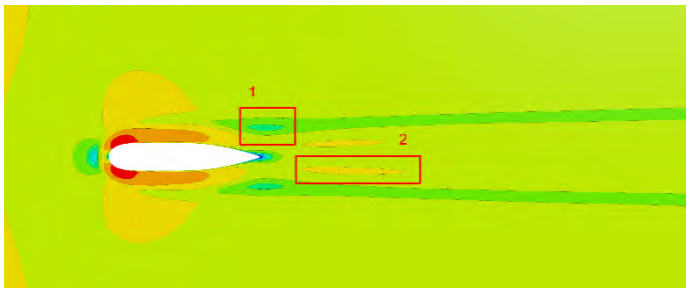


Figure 5. Selected local region

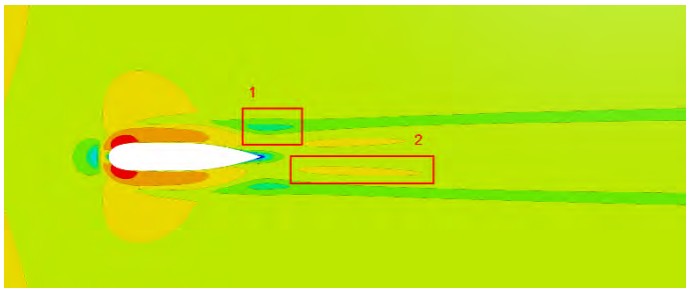
Table 1. Three sets of mesh information

Case	r ase size (m)	oe sh volume	$\overline{SAP}$
grid1	0.35	2,704,816	77.46
grid2	0.44	3,783,460	77.49(0.038%)
grid3	0.53	6,554,996	77.50(0.013%)

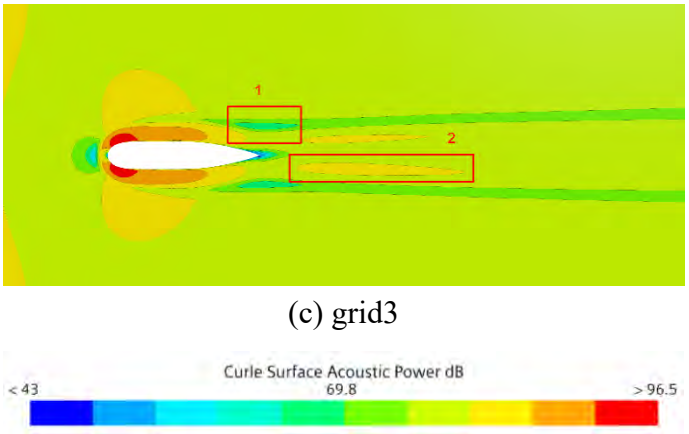
The distribution of sound source on the bottom surface is presented, comparing the results for different grid scales. As shown in the Fig. 6, with the increase in grid resolution, the structures corresponding to frames 1 and 2 become elongated. Additionally, since the simplified model has a symmetric structure, the results for Grid 2 exhibit the highest symmetry. Therefore, Grid 2 is selected for subsequent calculations after comprehensive consideration.



(a) grid1



(b) grid2



(c) grid3

Figure 6. Results corresponding to the three sets of grids

As shown in Tab. 1, with the increase in the number of grids, the calculation results initially decrease and then increase. This indicates that the calculation results do not consistently improve with an increasing number of grids. Therefore, by comparing the results of the three grid sets, the second set of grids is selected for subsequent calculations.

Result analysis and discussion

In this study, the position of the vortex generator, namely the horizontal coordinate x, vertical coordinate y, height h, and angle θ, are selected as design variables. H is computed to be 25 by Eq. (19), indicating the generation of 25 sets of computational samples through RSo. Due to the presence of the sail, the position of the vortex generator is constrained. When the vortex generator is far from the sail, it will not produce favorable effects. Therefore, the ranges of design variable values are as presented in Tab. 2. For ease of comprehension, one example is illustrated in the Fig. 7.

Table 2. Parameter ranges of the design variables

oa me	Range
horizontal coordinate x (m)	[50,80]
vertical coordinate y (m)	[30,50]
height h (m)	[0,30]
angle θ	[10,20]

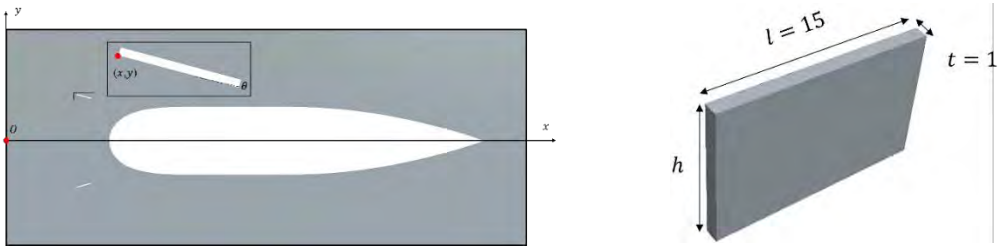
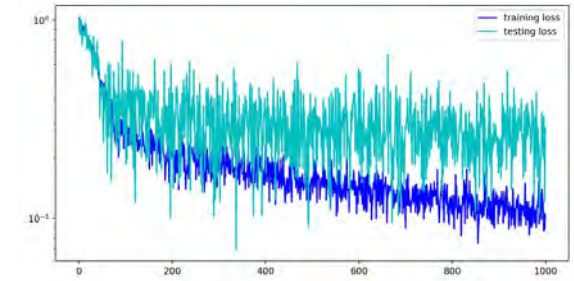


Figure 7. Installation region of the vortex generator

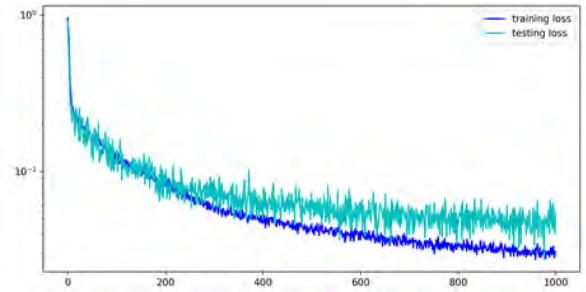
The computations for the 25 sets of computational samples were performed using CFD, and the results were input into Design Expert to obtain a four-variable quadratic regression equation as Eq. (25). To enhance the regression effectiveness, the intervals for the three length variables were translated to create symmetric intervals centered at zero. This equation was then utilized for pre-training to explore the impact of different initial sample sizes on the training results of Aoo . Three initial sample sizes were selected in this study: S=100, S=1000, and S=5000, with the training set accounting for 85% of the total and the testing set accounting for 15% of the total. The obtained training residual curves are illustrated in Fig. 7.

$$\begin{aligned}\overline{SAP} = {} & 76.95 - 0.23h + 0.064\theta - 0.013x_1 + 6.47\times10^{-3}x_2 \\ & - 2.62\times10^{-3}x_1h - 1.42\times10^{-3}x_2h + 0.01x_1\theta \\ & - 1.29\times10^{-3}x_2\theta + 2.57\times10^{-4}x_1x_2 - 4.8\times10^{-4}h^2 \\ & - 0.01\theta^2 + 9.24\times10^{-4}x_1^2 + 1.67\times10^{-3}x_2^2\end{aligned} \tag{25}$$

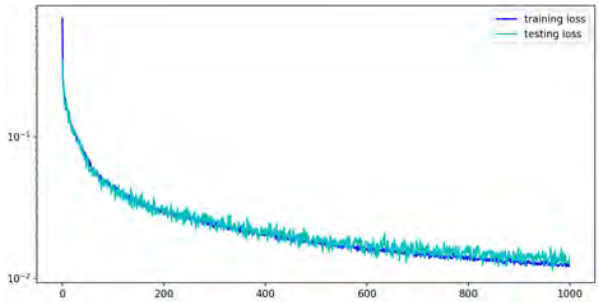
As shown in Fig. 8, with an increase in the number of samples, the oscillations in the residual curves are suppressed, and the errors in both the training and testing curves significantly decrease. When the total sample size reaches 5000, the training error and testing error almost overlap. Therefore, the training results corresponding to a total sample size of 5000 are selected as the pre-training model.



(a) S=100



(b) S=1000



(c) S=5000

Figure 8. Loss corresponding to different sample sizes

Twenty-five sets of computational samples were obtained using the OLHS method, and their values were computed via CFD. The 24 computational cases were input into the Ao o, using a leave-one-out validation approach, and the residual error curves for the training process were obtained. As shown in Fig. 9, the loss decreases rapidly when the training epochs are less than 50, and then decreases slowly for epochs greater than 50. The training loss does not exhibit significant oscillation, while the testing loss shows slight oscillations due to the small number of test cases. After 500 training epochs, the training loss is below 0.005 and the testing loss is below 0.01. Compared to Fig. 7(c), both losses have improved, confirming the feasibility of the method.

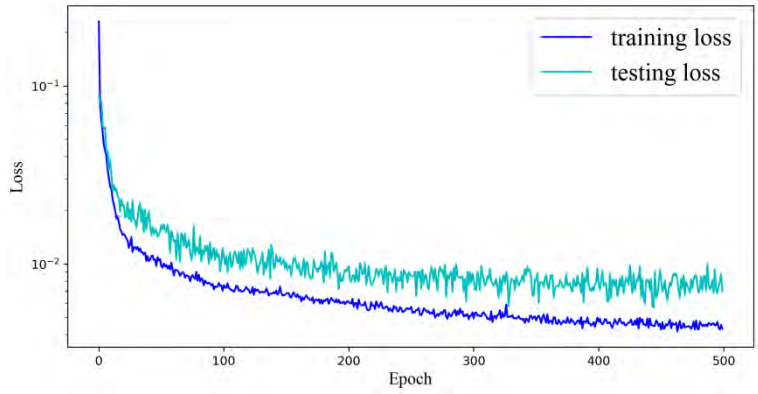


Figure 9. Loss curve

The training accuracy of the two error validation models was calculated by Eq. (23) and (24), as shown in Tab. 3. The table clearly demonstrates that this method meets the required computational accuracy.

Table 3. Prediction error

MAE	R^2
0.1124	0.9356

The highly accurate surrogate model obtained in the end is subjected to parameter optimization using a genetic algorithm, with the fitness function set as the reciprocal of the objective function. r inary encoding is employed for the optimization process. Tab. 4 presents the final optimization results. The optimal combination is validated through CFD, and the relative error ε between the predicted and computed values is calculated using Eq. (26).

According to Tab. 5, it is evident that the prediction error is within an acceptable range and is smaller compared to the target variables corresponding to the 25 sets of computational samples. Therefore, it can be considered that this case represents the optimal solution. Through calculation, it is determined that the installation of vortex generators results in a decrease of 0.18 dr in the average $\overline{SAP}$.

Table 4. Optimal combination of design variables

Parameters	$[h, x_1, x_2, \theta]$
Design variables	[10,50,48.36,30]

$$\varepsilon_i = \frac{G - G_{CFD}}{G_{CFD}} \times 100\% \quad (\mathrm{i} = 1,2,3) \tag{26}$$

where G represents the predicted value and G_{CFD} represents the computed value.

Table 5. Error analysis of the optimal solution

Design variables $[h, x_1, x_2, \theta]$	Computed value (dr)	Predicted value (dr)	Relative error ε
[18.5,69.1,53.6,20]	77.31	77.33	0.025%

Visualizing the computational results, comparing the flow field with and without vortex generators, and the distribution of surface acoustic sources. Plotting iso-surfaces of vorticity, cross-sectional velocity, and the distribution of acoustic sources on the bottom surface. Firstly, the vorticity cloud around the sail is presented, colored by velocity. The relevant variables have been dimensionless. r ased on Fig. 10(a), as the fluid flows towards the shroud rudder, it forms horseshoe vortices due to obstruction. Two symmetrical vortex legs can be observed at the rear of the rudder. As depicted in Fig. 10(b), after installing the vortex generator, the pre-existing low-pressure zones do not disappear but rather, two new low-pressure zones form around the vortex generator. The original horseshoe vortices are impeded and move upward, while those closer to the bottom are reduced. Flow separation phenomena weaken, and the length of the vortex legs decreases.

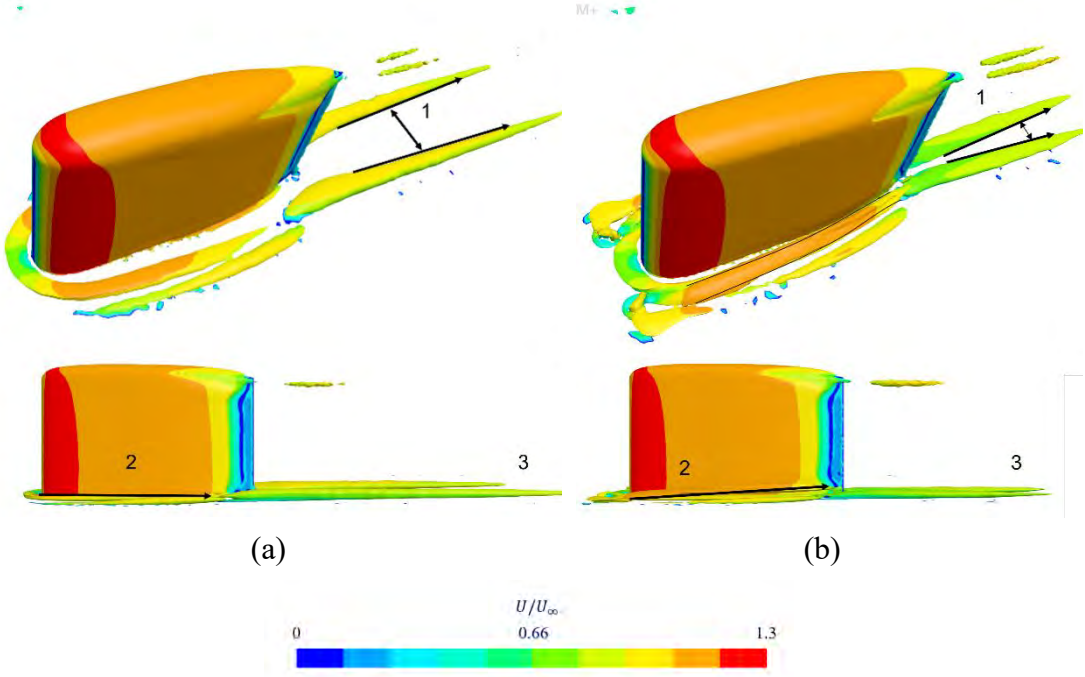


Figure 10. Q=20s^{-2} iso-surface

To illustrate the flow changes more clearly, velocity contour maps were generated for four cross-sections. The positions of these four cross-sections relative to the shroud rudder in the flow field are indicated in Fig. 11. In Fig. 12(a), a distinct low-velocity region can be observed in front of the intersection between the sail and the bottom, indicating the obstruction effect of the sail on the fluid. Fig. 12(b) shows that the fluid experiences localized acceleration as it passes the front end of the sail, accompanied by a change in flow direction. In Fig. 12(c), no flow separation is observed yet, but low-velocity horseshoe vortices near the wall can be seen. Fig. 12(d) reveals that flow separation results in a wake region with velocities lower than the average flow speed,

and the horseshoe vortices become more prominent.

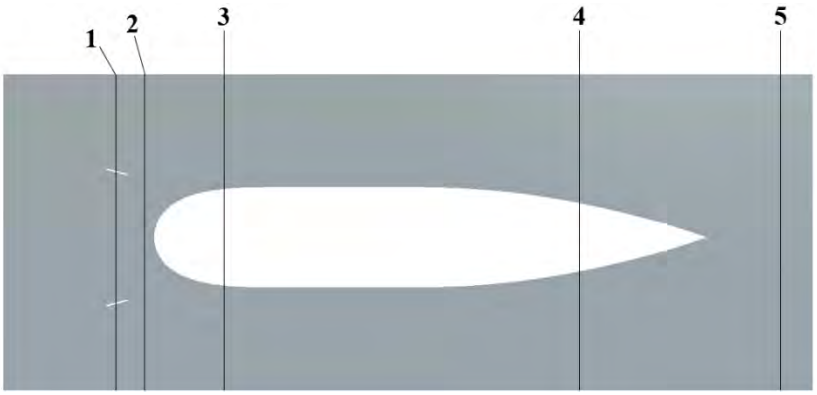
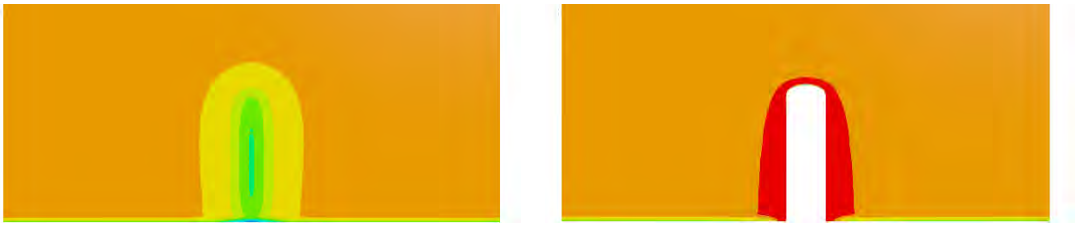


Figure 11. Cross-Sectional Relative Locations

According to Fig. 13(a), after the installation of the vortex generator, two new low-velocity regions appear, with the velocity outside the vortex generator being consistent with the far-field velocity. In Fig. 13(b), two vortex cores far from the bottom surface can be clearly observed. In Fig. 13(c), it can be seen that the two new vortex cores are closer to the shroud rudder, and the original horseshoe vortices are controlled. In Fig. 13(d), it is observed that the wake of the shroud rudder becomes more uniform, the horseshoe vortices are significantly suppressed, and two new low-velocity vortex cores exist within the boundary layer.

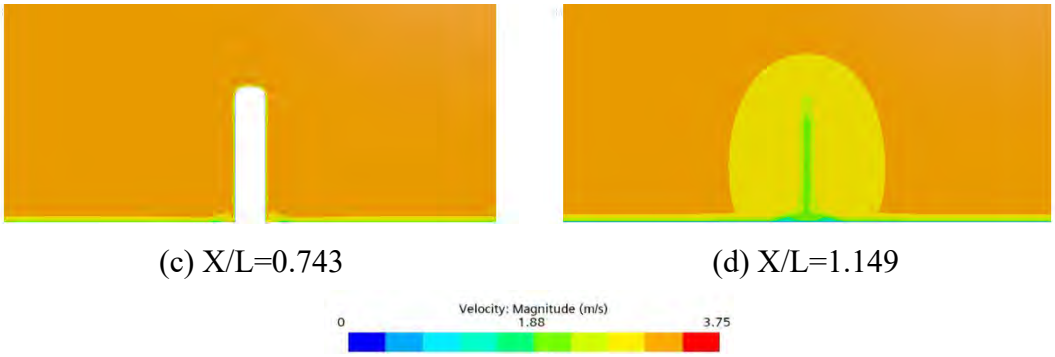


(a) X/L=-0.054 (b) X/L=0.122

(c) X/L=0.743 (d) X/L=1.149

Figure 12. Velocity contour maps of the original model flow field section

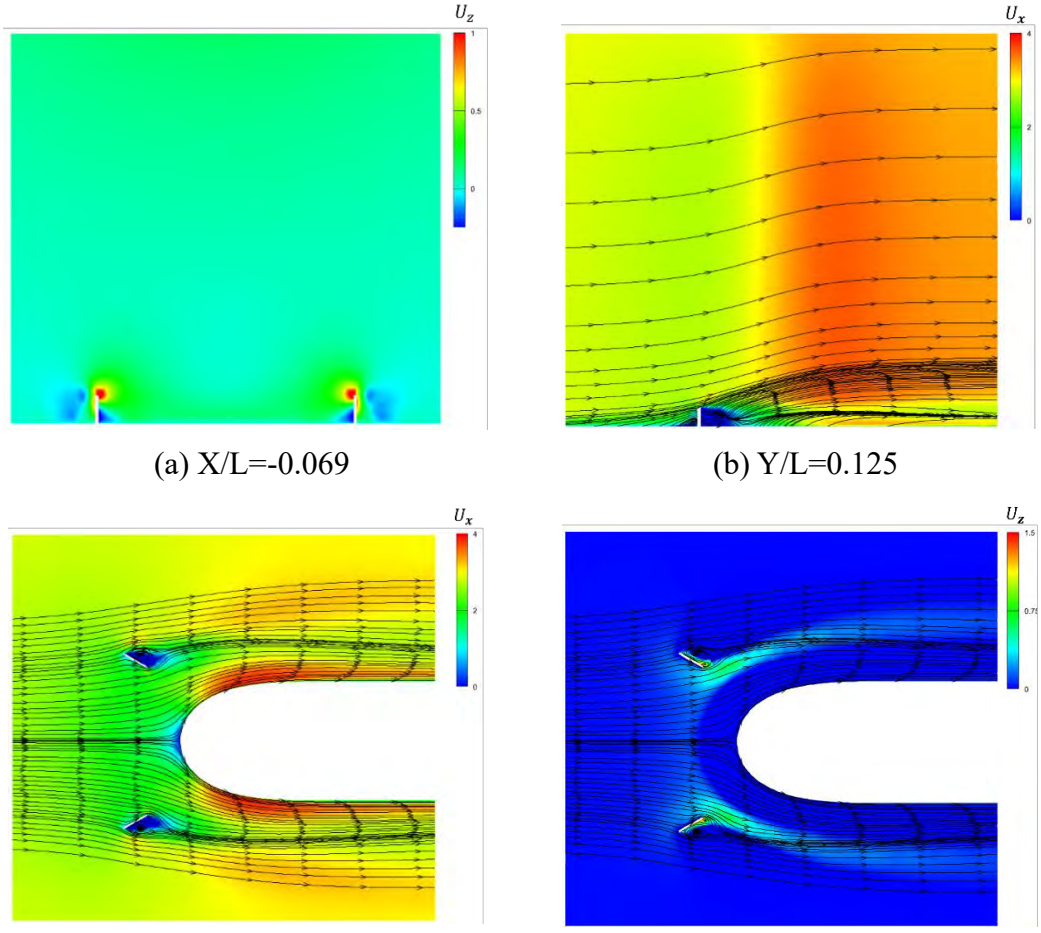
(a) X/L=-0.054 (b) X/L=0.122



(c) X/L=0.743 (d) X/L=1.149

Figure 13. Velocity contour maps of the flow field section with vortex generator

A detailed analysis of the vortex generator was conducted, demonstrating the flow characteristics in its vicinity. Fig. 14(a) and 14(d) illustrate U_z, showing significant positive z-directional velocity at the top and rear of the vortex generator, while negative U_z is observed at the bottom and outer sides, indicating the presence of vortices in these regions. Fig. 14(b) and 14(c) provide a more intuitive view of the flow trends, showing small vortices forming at the base of the vortex generator. The fluid exhibits an upward movement as it passes over the top of the vortex generator, leading to the formation of new vortices. According to Fig. 15, the installation of the vortex generator significantly mitigates flow separation.



(a) X/L=-0.069 (b) Y/L=0.125

(c) Z/L=0.016 (d) Z/L=0.016

Figure 14. Local Flow Details around the Vortex Generator[yz1]

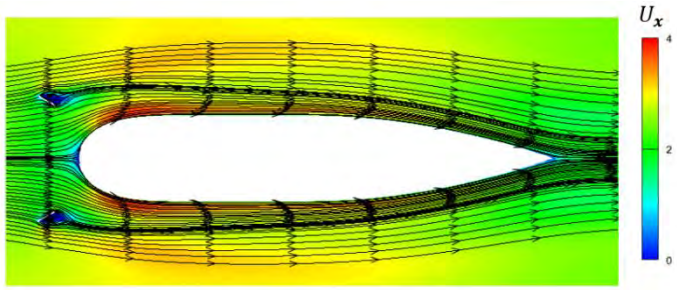


Figure 15. Fluid Flow over the Sail

The sound source map of the bottom surface is depicted, as shown in Fig. 16. As shown in Fig. 16(a), two mid-bottom sound source regions appear green and extend to the bottom of the image, with two essentially symmetrical ellipses observed in yellow. According to Fig. 16(b), after installing the vortex generator, the two elliptical mid-high sound bands disappear, and the distribution of surface sound sources becomes more uniform.

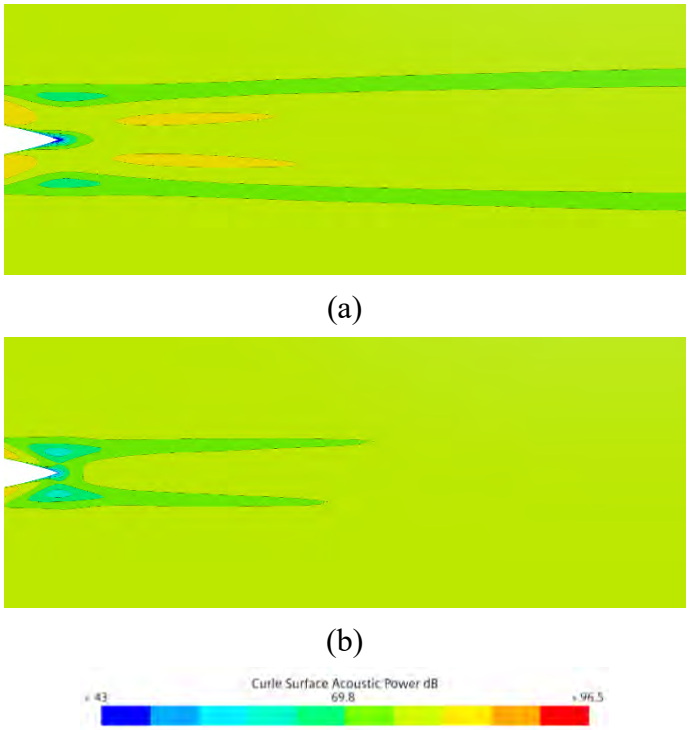


Figure 16. Schematic of $\overline{SAP}$

Conclusions and Future Perspectives

r y employing a broadband noise source model, this study investigates the suppression of horseshoe vortices and the improvement of acoustic sources due to the vortex generator. Sampling is conducted using the response surface methodology (RSo) and orthogonal Latin hypercube sampling (OLHS) methods, followed by regression to obtain pre-training equations. Subsequently, a high-precision surrogate model is obtained based on artificial neural networks (Ao o) . Finally, genetic algorithms (GA) are employed to optimize the parameters of the vortex generator. This study concludes with the following key findings:

(1) The vortex generator effectively improves the distribution of sound sources on the bottom surface.
(2) The vortex generator effectively improves the horseshoe vortices and makes the distribution of wake velocity more uniform.
(3) Constructing a high-precision surrogate model based on the concept of small-sample learning effectively mitigated overfitting during the training process.

Furthermore, since the current calculations are based on Reynolds-averaged equations and sound source estimation formulas, which may not comprehensively reflect the flow noise, this study will continue to conduct fine-scale simulations. Specifically, Large Eddy Simulation (LES) will be employed to compute the flow field, and the Ffowcs Williams-Hawkings (FWH) equation will be utilized to calculate the acoustic field.

Acknowledgments

This work is supported by the oa tional o atural Science Foundation of China (52131102), to which the authors are most grateful.

References

[1] Liu Z, Xiong Y., Tu C. oe thod to control unsteady force of submarine propeller based on the control of horseshoe vortex[J]. Journal of ship research, 2012, 56(01): 12-22.

[2] Liu Z, Xiong Y, Tu C. The method to control the submarine horseshoe vortex by breaking the vortex core[J]. Journal of Hydrodynamics, 2014, 26(4): 637-645.

[3] Liu G, Hao Z, r ie H, et al. Control mechanism of a vortex control baffle for the horseshoe vortex around the sail of a DARPA SUrOFF model[J]. Ocean Engineering, 2023, 275: 114166.

[4] Liu Y, Jiang H, Li Y, et al. Suppression of the Hydrodynamic o oise Induced by the Horseshoe Vortex through oe chanical Vortex Generators[J]. Applied Sciences, 2019, 9(4): 737.

[5] Thurman C S, Somero J R. Comparison of meta-modeling methodologies through the statistical-empirical prediction modeling of hydrodynamic bodies[J]. Ocean Engineering, 2020, 210: 107566.

[6] Rostamzadeh-Renani o, Rostamzadeh-Renani R, r aghoolizadeh o, et al. The effect of vortex generators on the hydrodynamic performance of a submarine at a high angle of attack using a multi-objective optimization and computational fluid dynamics[J]. Ocean Engineering, 2023, 282: 114932.

[7] Saghi H, Parunov J, oiku lić A. Resistance Coefficient Estimation for a Submarine's r are Hull oovi ng in Forward and Transverse Directions[J]. Applied Sciences, 2022, 12(21): 10953.

[8] Zhang L, Chen J T, Xiong F F, et al. oe ta-learning based multi-fidelity deep neural networks metamodel method[J]. Journal of o echanical Engineering, 58(1): 190-200.

[9] Leng J X, Feng Y, Huang W, et al. Variable-fidelity surrogate model based on transfer learning and its application in multidisciplinary design optimization of aircraft[J]. Physics of Fluids, 2024, 36(1).

[10] Agatonovic-Kustrin S, r eresford R. r asic concepts of artificial neural network (Aoo) modeling and its application in pharmaceutical research[J]. Journal of pharmaceutical and biomedical analysis, 2000, 22(5): 717-727.

[11] Serani A, Fasano G, Liuzzi G, et al. Ship hydrodynamic optimization by local hybridization of deterministic derivative-free global algorithms[J]. Applied Ocean Research, 2016, 59: 115-128.

[12] Lin Y, Yang Q, Guan G. Automatic design optimization of SWATH applying CFD and RSo mo del[J]. Ocean Engineering, 2019, 172: 146-154.

[13] Karolczuk A, Kurek o. Fatigue life uncertainty prediction using the oont e Carlo and Latin hypercube sampling techniques under uniaxial and multiaxial cyclic loading[J]. International Journal of Fatigue, 2022, 160: 106867.

[14] Liu Y, Xue Y, Chen Y, et al. Identification of nonparametric thermodynamic model and optimization of ocean thermal energy conversion radial inflow turbine[J]. Applied Energy, 2022, 321: 119348.

[15] Zhang Q, oiao W, Liu Q, et al. Optimized design of wind turbine airfoil aerodynamic performance and structural strength based on surrogate model[J]. Ocean Engineering, 2023, 289: 116279.

Acoustic Characteristics of Laminar-Turbulent Transition in a Flat Plate Based on PSE and FW-H

Bohan Xie[1*], Fuchang Zhou[2], Yuan Zhuang[1] and Decheng Wan[1†]

[1] Computational Marine Hydrodynamics Lab (CMHL), School of Naval Architecture, Ocean and Civil Engineering, Shanghai Jiao Tong University, Shanghai, China

[2] Wuhan Second Ship Design and Research Institute, Wuhan, China

[*]Presenting author: xiebh1997@sjtu.edu.cn
[†]Corresponding author: dcwan@sjtu.edu.cn

Abstract

In the present study, nonlinear parabolized stability equation (PSE) and wall-resolved large-eddy simulation (WRLES) are employed to simulate laminar-turbulent transition in a flat plate using the solver OpenFOAM. Fluctuations of boundary layer displacement thickness, pressure and velocity in laminar, transition and turbulence regions are compared in detail. By combining nonlinear PSE and porous-surface Ffowcs-Williams and Hawkings (FW-H) method, the acoustic characteristics in different regions of the flat plate are analyzed. Our results have shown that the fluctuations of boundary layer displacement thickness and pressure in transition region begin to increase. Besides, the fundamental wave and subharmonic wave introduced at the inlet boundary have significant influence on the spectral curves of flow field quantities and radiated noise.

Keywords: Laminar-turbulent Transition, Parabolized Stability Equation, Large-eddy Simulation, Radiated Noise

Introduction

Laminar-turbulent transition in a boundary layer has been extensively studied in recent decades. Laminar-turbulent transition occurs when an ambient disturbance enters the boundary layer and excites unstable disturbance waves, and the boundary layer eventually breaks into turbulence with the nonlinear interactions of disturbance waves [1]. The flow in the transition region is significantly different from that in laminar region and turbulent region. It has been shown that the resistance of vehicles and the operation efficiency of propulsion systems are closely related to the transition onset position and transition region length [2]. However, the relevant research on the characteristics of radiated noise in transition region is still scarce.

Several numerical approaches have been developed to simulate laminar-turbulent transition process. Direct numerical simulation (DNS) has the highest accuracy in resolving small-magnitude disturbances in boundary layer. By introducing a pair of oblique waves at low amplitudes in a supersonic flat-plate boundary layer, Mayer (2011) demonstrated that oblique breakdown was a viable path to sustained turbulence [3]. Subbareddy (2012) discussed the development of mean profiles in the transitional and fully turbulent regions by DNS and pointed out a “two-layer” structure in transitional regions [4]. Sayadi (2013) simulated complete transition to turbulence of both H-type and K-type, and discussed in detail the differences between the two types of transition [5]. Drikakis (2021) compared the transition acoustic loading at Mach 6 of DNS and implicit LES, it was found that DNS had advantages in predicting transition onset positions and high frequency peaks [6].

Due to the high computational cost of DNS, Linear stability theory (LST) is an alternative to simulating laminar-turbulent transition [7]. LST can be used to describe the linear growth of unstable modes. Jee (2018) captured H-type breakdown generated by nonlinear interactions between the fundamental wave and the subharmonic oblique wave by LST [8]. Danvin (2018) demonstrated the performances of using LST in predicting the evolution of Mack's second mode for compressible hypersonic flows [9]. Bae (2023) obtained rich high-fidelity computations by combining DNS and LST, and revealed that high spanwise wavenumber modes were associated with the final breakdown to the turbulent flow [10].

However, LST ignores the nonlocal and nonparallel effects in the boundary layer and thus is not suitable for simulating the nonlinear growth of the disturbances. PSE proposed by Herbert (1997) well solved the problem [11], and was capable of simulating laminar-turbulent transition economically. Lozano (2018) reproduced downstream turbulence by modeling pretransitional region with PSE [12].Kim (2019) managed to reproduce the subharmonic resonance by nonlinear PSE, and pointed out that the PSE inlet located downstream could effectively reduce computational cost [13]. Based on PSE, Olichevis (2021) proposed a robust transition framework to conduct PSE calculations for transition onset locations [14].

In the present paper, WRLES and nonlinear PSE are adopted to simulate laminar-turbulent transition in a flat plate. The evolutions of flow field in the laminar, transition and turbulence regions are analyzed. Furthermore, the acoustic characteristics in different regions are evaluated by FW-H method.

Numerical Methods

Large-eddy simulation

LES separates large-scale and small-scale eddies in the flow field by filtering functions. Large-scale eddies are directly simulated, and small-scale eddies are resolved by sub-grid scale models [15]. The spatial filtered governing equations are expressed as follows:

$$\frac{\partial \bar{u}_i}{\partial x_i} = 0 \tag{1}$$

$$\frac{\partial \bar{u}_i}{\partial t} + \frac{\partial}{\partial x_j}\left(\bar{u}_i \bar{u}_j\right) = -\frac{1}{\rho}\frac{\partial \bar{p}}{\partial x_i} + \nu \frac{\partial^2 \bar{u}_i}{\partial {x_j}^2} - \frac{\partial \tau_{ij}}{\partial x_j} \tag{2}$$

where the subscripts i and j represent streamwise, normal and spanwise directions, the symbol $\bar{\cdot}$ denotes the spatial filtering, p is the pressure, ν is the kinematic viscosity of fluid, $\tau_{ij} = \bar{u}_i \bar{u}_j - \overline{u_\iota u_j}$ is the sub-grid stress to describe the interactions between the large-scale eddies and small-scale eddies.

The wall-adapting local eddy-viscosity (WALE) model is adopted in the present study. WALE is a sub-grid scale model based on the square of the velocity gradient tensor, which takes both the shear stress tensor and the rotation tensor into account [16]. The expression of sub-grid viscosity ν_t of WALE is as follows:

$$\nu_t = (C_w \Delta)^2 \frac{(S_{ij}^d S_{ij}^d)^{3/2}}{(\bar{S}_{ij}\bar{S}_{ij})^{5/2} + (S_{ij}^d S_{ij}^d)^{5/4}} \tag{3}$$

where C_w is the model coefficient, Δ is the grid filter width defined by cell volume, S_{ij}^d is the tensor defined as follows:

$$S_{ij}^d = \frac{1}{2}\left(\frac{\partial \bar{u}_i}{\partial x_k}\frac{\partial \bar{u}_k}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_k}\frac{\partial \bar{u}_k}{\partial x_i}\right) - \frac{1}{3}\delta_{ij}\frac{\partial \bar{u}_k}{\partial x_k}\frac{\partial \bar{u}_k}{\partial x_k} \tag{4}$$

Parabolized stability equation

The variables $\breve{\phi}$ in the flow field can be decomposed into the following form:

$$\breve{\phi} = \Phi + \phi \tag{5}$$

where Φ is the mean quantities and ϕ is the disturbances.

The basic idea of PSE is to decompose each mode in a disturbance ϕ into a slow-varying shape function and a fast-varying wave function [17]. Assuming that the disturbance ϕ is periodic in both spanwise and time, it can be expressed in the following Fourier series form:

$$\phi(x,y,z,t) = \sum_{m=-N_m}^{N_m} \sum_{n=-N_n}^{N_n} \hat{\phi}_{(m,n)}(x,y) \left[\exp i \int_{x_0}^{x} \alpha_{(m,n)}(\xi) d\xi + in\beta z - im\omega t \right] \tag{6}$$

where x, y, z indicate streamwise, normal and spanwise directions respectively, $\hat{\phi}_{(m,n)}$ is the shape function of Fourier mode (m,n), $\alpha_{(m,n)}$ is the streamwise complex wave number of Fourier mode (m,n), subscripts m and n are the temporal and spanwise wave modes respectively, β is the spanwise wave number, ω is the angular frequency.

In the present study, nonlinear PSE is used to calculate nonlinear growth of disturbances at the early stage, and the downstream nonlinear interactions and transition to the turbulence are simulated by LES instead.

Ffowcs-Williams Hawkings equation

Noise generated by fluid was first equivalent to sound sources by the acoustic analogy proposed by Lighthill [18]. Based on the Lighthill's theory, Ffowcs-Williams and Hawkings (1969) derived the FW-H equation, which extended the Lighthill's theory to the case of considering a rigid boundary with arbitrary motion in the flow field [19]. The original FW-H equation is shown as follows.

$$\odot^2 p' = \frac{\partial}{\partial t}[\rho_0 v_n \delta(f)] - \frac{\partial}{\partial x_i}[p n_i \delta(f)] + \frac{\partial^2}{\partial x_i \partial x_j}[H(f) T_{ij}] \tag{7}$$

$$T_{ij} = \rho u_i u_j - \sigma_{ij} + (p' - c^2 \rho')\delta_{ij} \tag{8}$$

where $\odot^2 = \frac{1}{c^2}\frac{\partial^2}{\partial t^2} - \nabla^2$ is the D'Alembert operator, $\rho' = \rho - \rho_0$ is the density perturbation, c is the velocity of sound in the fluid, ρ_0 is the density of the fluid, v_n is the normal velocity of the integral surface, T_{ij} is the Lighthill's stress tensor, σ_{ij} is the viscous stress tensor, δ_{ij} is the Kronecker symbol, $\delta(f)$ is the Dirac function, $H(f)$ is the Heaviside function.

In the present study, the porous-surface FW-H method proposed by Di Francescantonio (1997) is applied rather than the original one. The reason will be mentioned later. The porous-surface FW-H equation can be expressed as below [20]

$$\odot^2 p' = \frac{\partial}{\partial t}[\rho_0 U_n \delta(f)] - \frac{\partial}{\partial x_i}[L_i \delta(f)] + \frac{\partial^2}{\partial x_i \partial x_j}[H(f) T_{ij}] \tag{9}$$

$$U_n = \left(1 - \frac{\rho}{\rho_0}\right) v_n + \frac{\rho u_n}{\rho_0} \tag{10}$$

$$L_i = p\delta_{ij} n_j + \rho u_i (u_n - v_n) \tag{11}$$

Farassat's formulation 1A converted FW-H equation into integral form as follows [21]

$$p'_S(x,t) = p'_T(x,t) + p'_L(x,t) \tag{12}$$

$$4\pi p'_T(x,t) = \int_{f=0} \left[\frac{\rho_0 \dot{v}_n}{r(1-Mr)^2} + \frac{\rho_0 v_n \hat{r}_i \dot{M}_i}{r(1-Mr)^3}\right]_{ret} dS + \int_{f=0} \left[\frac{\rho_0 c v_n (M_r - M^2)}{r^2(1-Mr)^3}\right]_{ret} dS \tag{13}$$

$$4\pi p'_L(x,t) = \int_{f=0} \left[\frac{\dot{p}cos\theta}{cr(1-Mr)^2} + \frac{\hat{r}_i \dot{M}_i p cos\theta}{cr(1-Mr)^3}\right]_{ret} dS + \int_{f=0} \left[\frac{p(cos\theta - M_i n_i)}{r^2(1-Mr)^2} + \frac{(M_r - M^2)pcos\theta}{r^2(1-Mr)^3}\right]_{ret} dS \tag{14}$$

where $p'_S(x,t)$ is the total surface term, $p'_T(x,t)$ is the thickness surface term, $p'_L(x,t)$ is the loading surface term, $1-Mr$ is the Doppler factor, $r = |x-y|$ is the distance from the sound source to the observer, subscript ret represents the retarded time $t - r/c$.

The sound pressure level (SPL) is defined as

$$SPL = 20\log_{10}(p/p_{\mathrm{ref}}) \tag{15}$$

where p is the acoustic pressure fluctuations and p_{ref}=1 μPa is the reference pressure for water. To represent the total level of SPL in a specific frequency band, the overall sound pressure level (OASPL) is defined as

$$OASPL = 20\log_{10}\left(\sum_{i=1}^{N} 10^{SPL_i/10}\right) \tag{16}$$

where SPL_i is the SPL value corresponds to the frequency i, N is the number of frequency points in the frequency band.

The current FW-H code in OpenFOAM references Epikhin's work [22].

Numerical setup

In this paper, LES coupled with nonlinear PSE is used to simulate the laminar-turbulent transition in a flat plate. The current computational domain is shown in Fig. 1, where $R=\sqrt{Re_x}=\sqrt{\frac{U_\infty x}{\nu}}$ denotes the nondimensional streamwise position. The freestream velocity U_∞=1 m/s and the kinematic viscosity of the fluid $\nu=6.25\times10^{-6}$ m^2/s. The upstream laminar simulation is conducted between $0\leq R<400$, where R=400 is the outlet for the laminar region as well as the inlet for the turbulence region. Flow field information of both laminar simulation and nonlinear PSE calculation are input at R=400. Laminar-turbulent transition and full turbulence are resolved at turbulence region between $400\leq R<1000$. Sponge region with $1000\leq R\leq1300$ is set to reduce the influence of outlet boundary on the flow. The boundary layer thickness δ_0=0.01 m at R=400 is used to undimensionalize the length scale. The length, height and the width of the computational domain are $L_x=1050\delta_0$, $L_y=300\delta_0$ and $L_z=12\delta_0$ respectively.

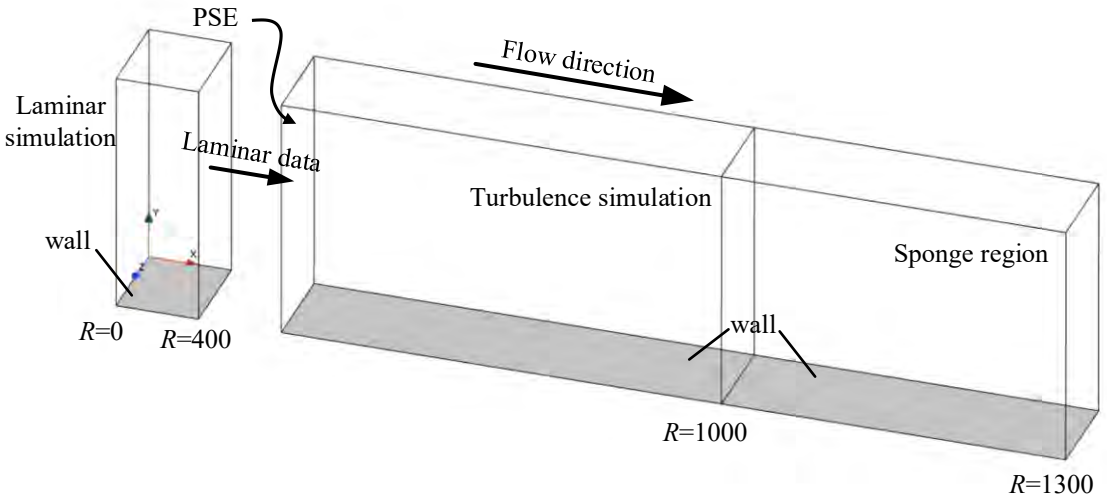


Fig. 1 Computational domain

According to Kachanov's experiment [23], a fundamental wave with mode (2,0) of $F=2\pi f\nu/U_\infty^2=124\times10^{-6}$ ($f\delta_0/U_\infty=3.2\times10^{-2}$), and a pair of subharmonic wave with mode (1,1) of $\frac{1}{2}F$ ($f\delta_0/2U_\infty=1.6\times10^{-2}$) are introduced at R=400. The streamwise wave number of the 2D Tollmien-Schlichting (TS) wave is $\alpha_{(2,0)}=2\pi\delta_0/\lambda_x=0.5418$, the spanwise wavenumber of the subharmonic wave is $\beta_{(1,1)}=2\pi\delta_0/\lambda_z=0.5206$, and the angular frequency of the TS wave is $\omega_{(2,0)}=2\pi f\delta_0/U_\infty=0.1984$.

Hexahedral grids with a total number of 2.916×10^7 are employed for the current simulation. The streamwise and spanwise scales of grids are isotropic. The normal scale of grids increases gradually as the distance to the boundary increases. Detailed grid parameters are shown in Tab. 1. The superscript + represents nondimensional by friction velocity u_τ and viscosity ν. The timestep used in the current simulation is $\Delta t_1U_\infty/\delta_0=0.1$ for over $T_1U_\infty/\delta_0=4000$ to obtain a steady flow, and is adjusted to $\Delta t_2U_\infty/\delta_0=0.005$ for over $T_2U_\infty/\delta_0=500$ to record unsteady flow information like pulsating pressure and noise.

Tab. 1 Grid parameters

Parameters	Laminar Simulation	Turbulence Simulation	Sponge Region
Δx^+	9.62	9.21	901.20
$\Delta y^+_{\min}$	0.75	0.81	0.53
$\Delta y^+_{\max}$	767.92	824.98	536.15
Δz^+	9.07	9.78	6.35
N_x	500	3000	60
N_y	128	128	128
N_z	64	64	64
$N_{\text{total}}\times10^6$	4.096	24.576	0.492

Numerical Simulations

The instantaneous vortex structure displayed based on the Q-criterion [24] is shown in Fig. 2. At the upstream region, 2D TS wave is the dominant component. As the flow moves downstream, the amplitude of the subharmonic wave gradually increases to the same order of

the fundamental wave, leading to the spanwise fluctuations in the flow field. The three dimensional instability promotes the evolution and interaction of vortex structures. As a consequence, turbulent spots and gamma vortex appear, which marks the beginning of the transition to turbulence. Turbulence spots continue to grow and develop downstream and merge with other turbulence spots, and eventually forming a fully developed turbulent flow [25].

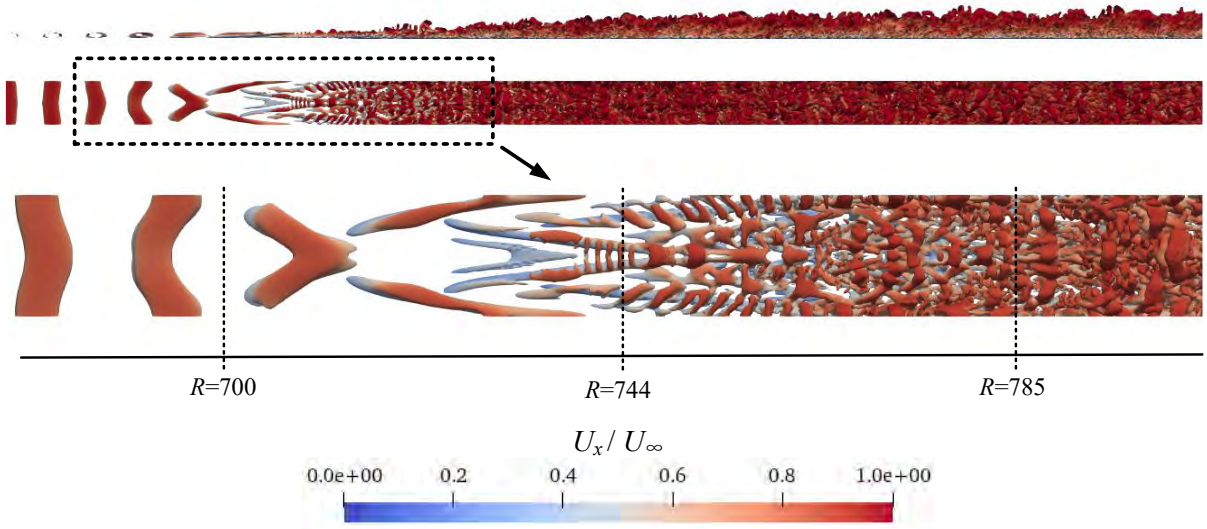


Fig. 2 Isosurface of Q=5 s^{-2} colored by nondimensional streamwise velocity

The skin friction coefficient C_f, defined as $C_f=\frac{\tau_w}{\frac{1}{2}\rho U_\infty^2}$ where τ_w is the mean wall shear stress, is an important parameter to describe the transition from laminar to turbulence. The streamwise distribution C_f of the current simulation compared with previous studies is shown in Fig. 3 [5], [12], [13]. The evolution of C_f shows that PSE coupled with LES method is able to resolve laminar-turbulent transition process accurately. It is generally believed that the location where C_f curve derivates from the laminar solution represents the onset of transition, and the transition ends at the peak of C_f curve. According to the current results, R=700 and R=785 are used to represent onset and end of laminar-turbulent transition respectively. Thus, the length of the transition region is $\Delta x_{tr}=79\delta_0$.

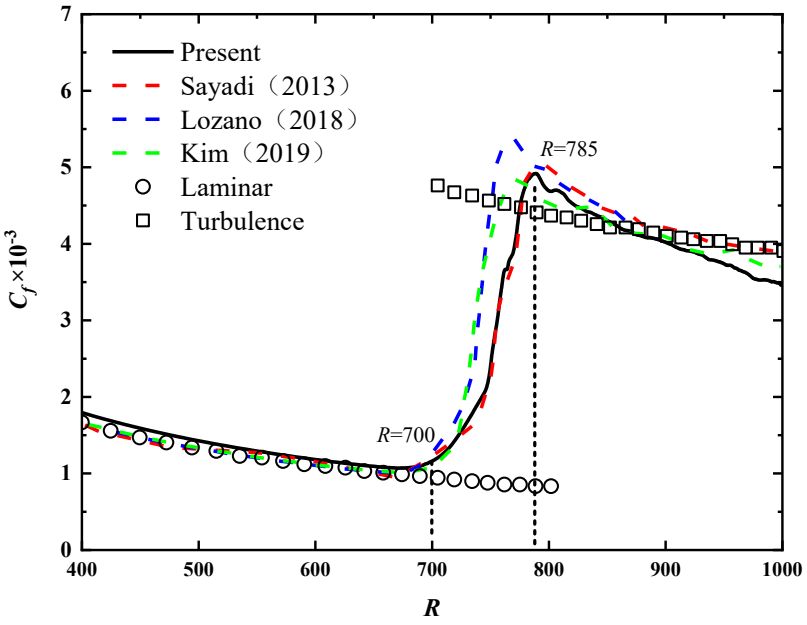


Fig. 3 Streamwise distribution of C_f in the flat plate

Three FW-H porous surfaces are employed to predict the radiated noise level in laminar, transition and turbulence regions, as shown in Fig. 4. All FW-H sound source regions are $79\delta_0$ in length, $30\delta_0$ in height and $30\delta_0$ in width, to ensure that each sound source region has the same resolution and number of grid cells. Three observers (M1, M2, M3) are located $100\delta_0$ from the wall, and at the center of each FW-H sound source region in streamwise and spanwise directions respectively. Each observer only corresponds to the FW-H porous surface directly below to avoid the influence of the sound propagation distance. An additional observer N1 located $1000\delta_0$ from the onset of transition corresponds to the porous surface at transition region is used to verify the simulation results of radiated noise in transition region in a flat plate.

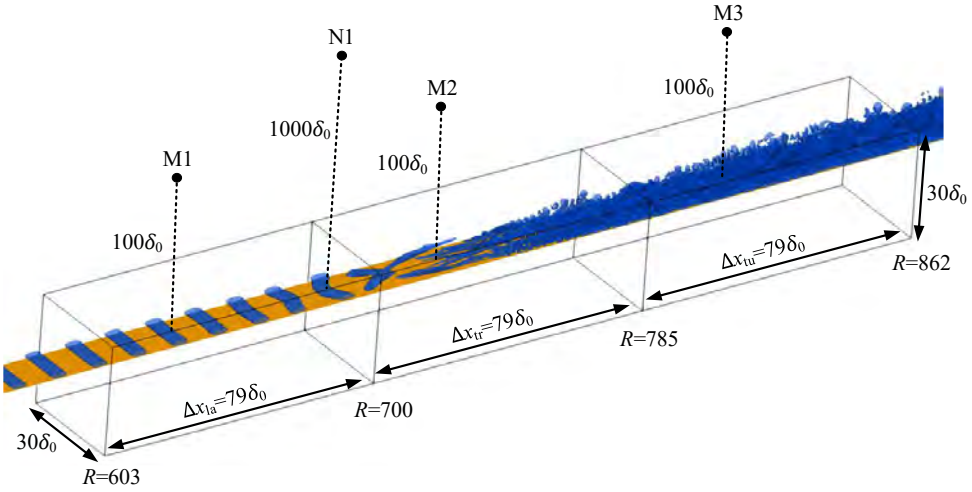


Fig. 4 Schematic diagram of the radiated noise prediction

Lauchle (1981) derived the empirical formula for the power spectral density (PSD) of the far-field acoustic pressure radiated per unit spanwise width of boundary layer in the transition region of a flat plate by equating the displacement thickness fluctuation to a monopole sound source [26]. The FW-H method based on the stationary surface cannot take monopole noise into account, so it is not suitable for predicting the radiated noise in the transition region. The comparison between current porous FW-H method and Lauchle's empirical formula at observer N1 is shown in Fig. 5, where ω is undimensionalized by δ_0 and U_∞. Considering that the empirical formula is idealized in some aspects, the present comparative results are generally in good agreement. This proves that porous FW-H method is able to accurately predict the radiated noise in the transition region.

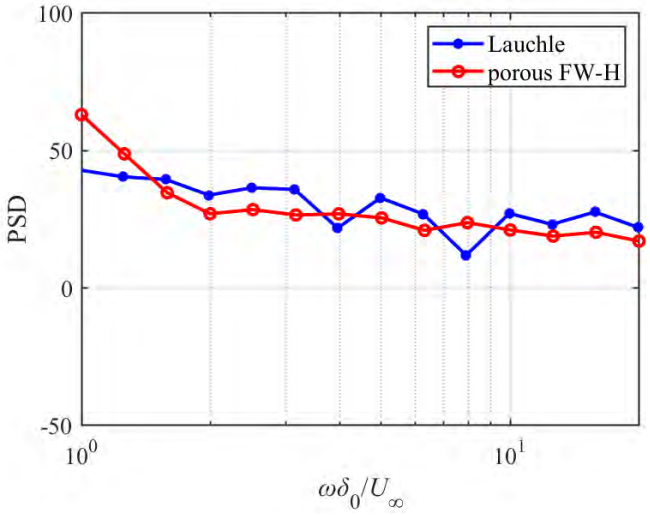
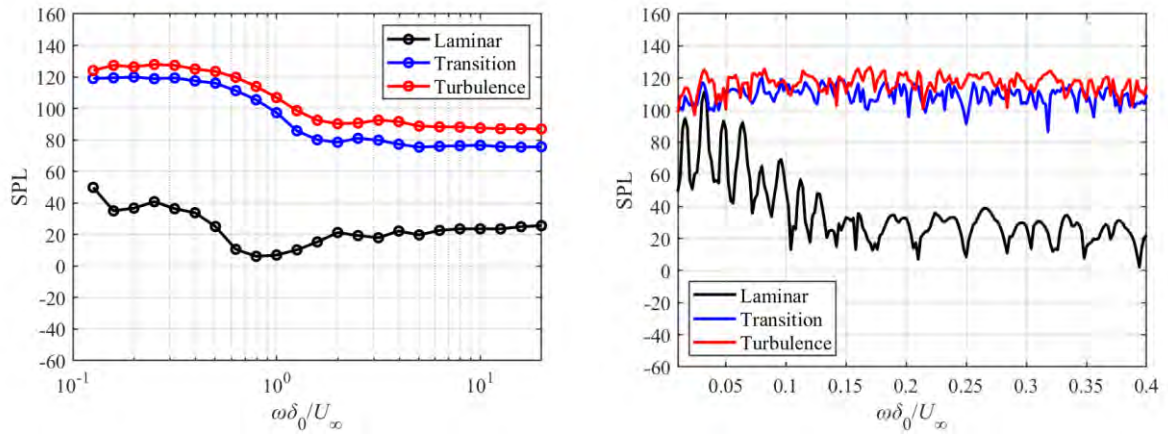


Fig. 5 Comparison of PSD between Lauchle's empirical formula and current porous FW-H method in the transition region on the plate

To further study the radiated noise characteristics on the flat plate, Fig. 6 provides comparisons of SPL in laminar, transition and turbulence regions. In Fig. 6 (a), one-third octave spectral of SPL from $0.1<\omega\delta_0/U_\infty<20$ indicate that noise generated in laminar region is much lower than that in transition region and turbulence region in each frequency band. SPL curves in transition region and turbulence region have similar frequency variation regulation, and noise in the turbulence region is slightly greater than that in the transition region. Since the high SPL amplitude parts in each region are concentrated in the low frequency segment, Fig. 6 (b) compares continuous spectral of SPL from $0.01<\omega\delta_0/U_\infty<0.4$ in different regions. The low frequency noise of transition region and turbulence region is almost of similar magnitude without obvious SPL peaks. However, the extremely low frequency ($\omega\delta_0/U_\infty<0.128$) noise in the laminar region exhibits significant spectral line characteristic with $\Delta\omega\delta_0/U_\infty=1.6\times10^{-2}$, which is very close to the frequency of subharmonic wave introduced at the inlet. When the $\omega\delta_0/U_\infty$ is higher than 0.16, the frequency interval of adjacent SPL peaks is approximately doubled in the laminar region. This phenomenon seems to suggest that the frequency features of radiated noise in the laminar region are significantly affected by TS wave and subharmonic wave. Flows in transition and turbulent regions are more intense, resulting in more complex sound sources of radiated noise. Therefore, the frequencies corresponding to TS wave and subharmonic wave are no longer dominant in radiated noise.



(a) One-third octave spectral **(b) Continuous spectral**

Fig. 6 Comparisons of SPL in different regions on the plate

In order to explore the relationship between flow field characteristics and radiated noise in different regions of the flat plate, the different regions of the flat plate are divided into four parts with equal area, as shown in Fig. 7. The changes of pulsating pressure, velocity and boundary layer displacement thickness δ^* are analyzed at the positions marked in the figure. δ^* is defined as follows

$$\delta^* = \int_0^\delta (1-\frac{u}{U_\infty})\mathrm{d}y \tag{17}$$

where u is the local streamwise velocity, δ is the boundary layer thickness defined as the distance from the boundary where u reaches 99% of U_∞.

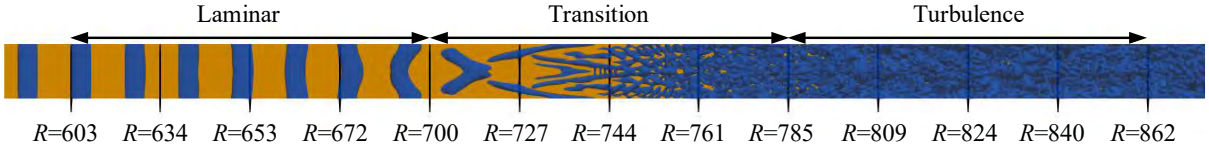
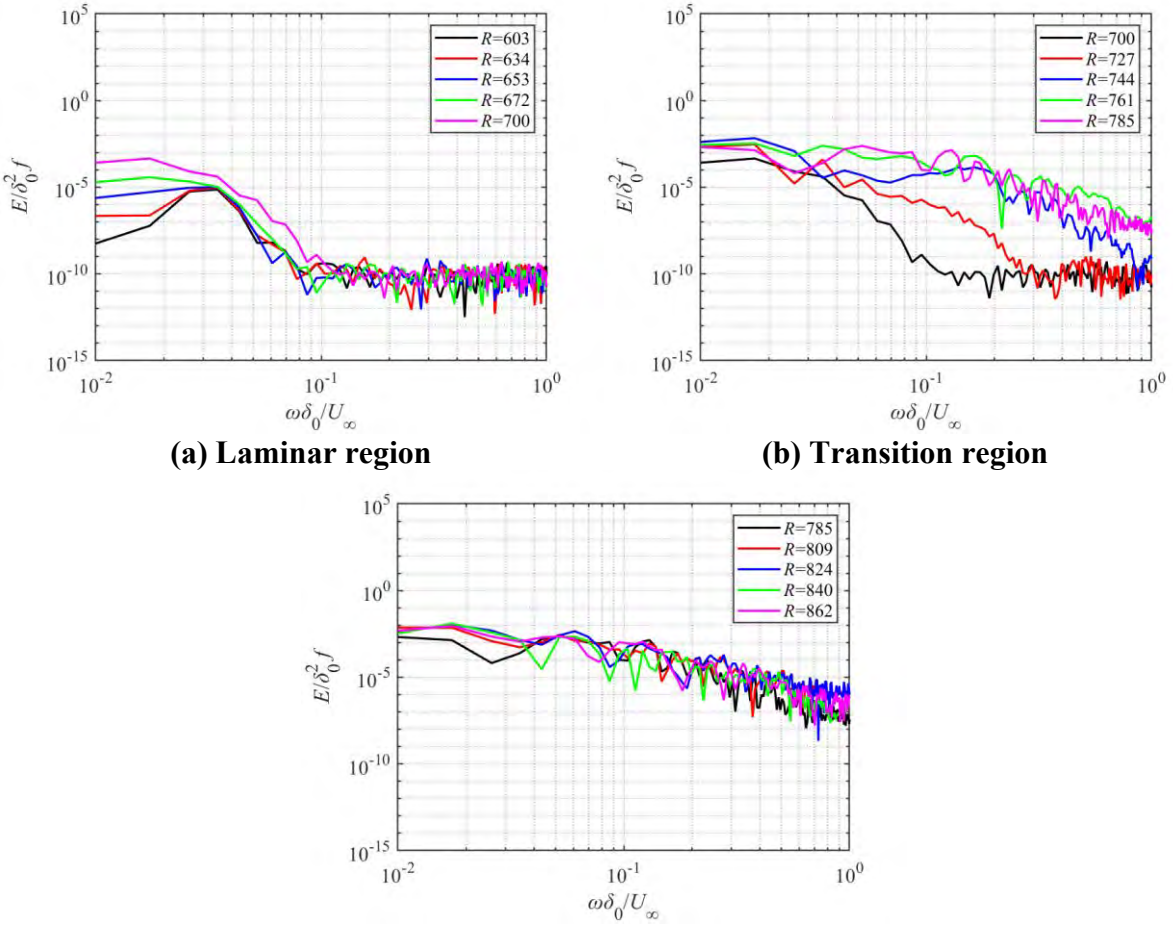


Fig. 7 Typical locations of different regions on the plate

Comparisons of PSD of boundary layer displacement thickness δ^* are shown in Fig. 8. The PSD results are nondimensionalized by δ_0 and f, where f is the frequency of fundamental wave satisfying $f\delta_0/U_\infty=3.2\times10^{-2}$. Both in the laminar region and turbulence region, the PSD of δ^* hardly changes in different streamwise locations. In the transition region, PSD value increases significantly with the increase of R in the frequency band $\omega\delta_0/U_\infty>0.04$. For all the typical locations, PSD curves reach peaks around $\omega\delta_0/U_\infty=1.6\times10^{-2}$ and $\omega\delta_0/U_\infty=3.2\times10^{-2}$. These results indicate that the fluctuations of boundary layer displacement thickness in the transition region begin to intensify, which confirms Lauchle's hypothesis [26]. In addition, TS wave and subharmonic wave introduced at the inlet boundary could affect fluctuations of the boundary layer displacement thickness.



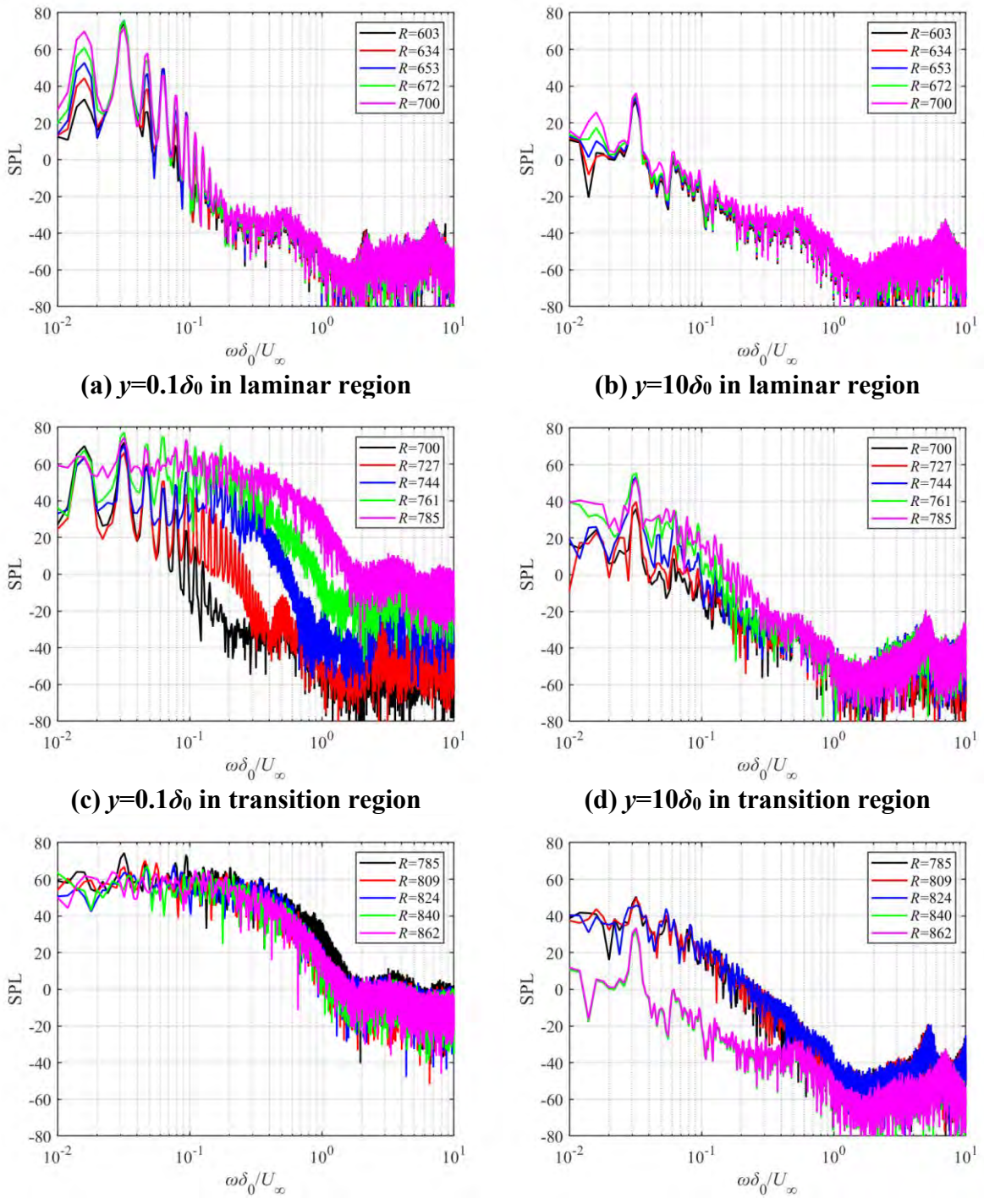
(a) Laminar region **(b) Transition region**

(c) Turbulence region

Fig. 8 PSD of boundary layer displacement thickness δ^* in different regions on the plate

Pressure and normal velocity are monitored in the typical streamwise locations in Fig. 7 at $y=0.1\delta_0$ (in the near-wall region) and $y=10\delta_0$ (at the porous FW-H surface, in the freestream). Comparisons of SPL curves of pressure nondimensionalized by $p_{ref}=1$ μPa are shown in Fig. 9. In the laminar region and early transition region at $y=0.1\delta_0$, several peaks in SPL curves with the frequency interval of $\Delta\omega\delta_0/U_\infty=1.6\times10^{-2}$ are observed at low frequency. At the downstream locations, the pulsating pressure has a higher order of magnitude throughout the low frequency. Besides, the pulsating pressure gradually increases from the onset of transition the full turbulence is developed. These phenomena indicate that the instability modes imposed by nonlinear PSE has influence on the pre-transition and early transition regions. But it seems that these subharmonic and higher harmonic waves only affect the pulsating pressure in the near-wall region, due to only the peak of SPL curves corresponding to the frequency of TS wave is detected in the freestream. According to the FW-H equation, the fluctuations of pressure can be

regarded as a dipole sound source [19]. The present numerical results show that this dipole sound source is mainly concentrated in the near-wall region, and is closely related to the disturbance modes.



(a) y=0.1δ_0 in laminar region **(b) y=10δ_0 in laminar region**

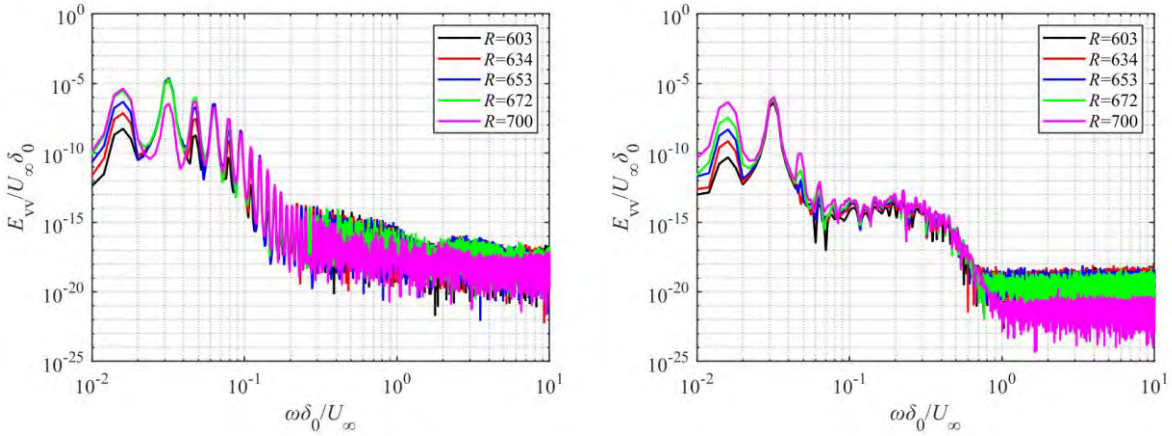
(c) y=0.1δ_0 in transition region **(d) y=10δ_0 in transition region**

(e) y=0.1δ_0 in turbulence region **(f) y=10δ_0 in turbulence region**

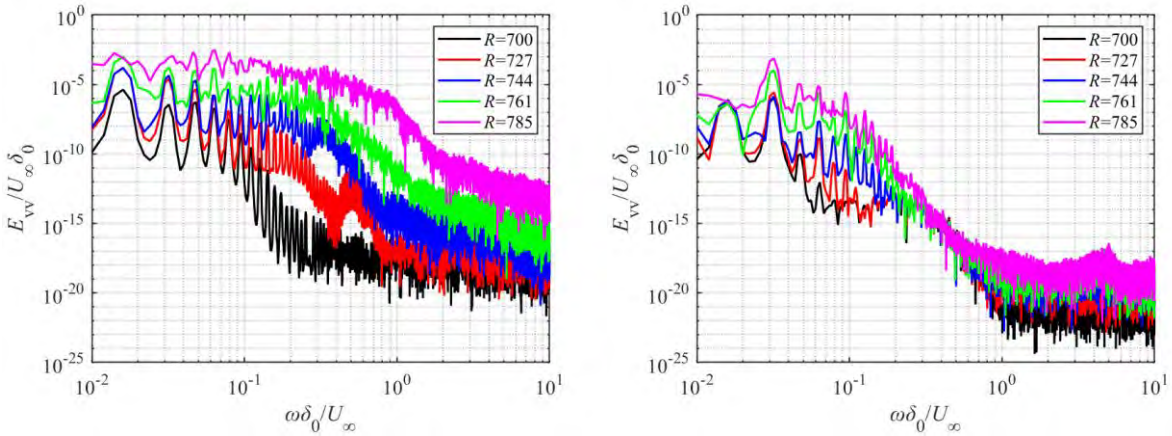
Fig. 9 SPL of pressure in different regions on the plate

The normal velocity at y=0.1δ_0 and y=10δ_0 represents the expansion of the boundary layer to the outside and the normal velocity through the porous FW-H surfaces, respectively. The PSD curves of normal velocity nondimensionalized by U_∞ and δ_0 in different regions are shown in Fig. 10. In the near-wall region, the PSD curves of normal velocity have similar variation trend with SPL curves of pulsating pressure, including the frequency characteristics. This proves once again that the influence of the disturbed modes on the flow field is comprehensive. Fluctuations

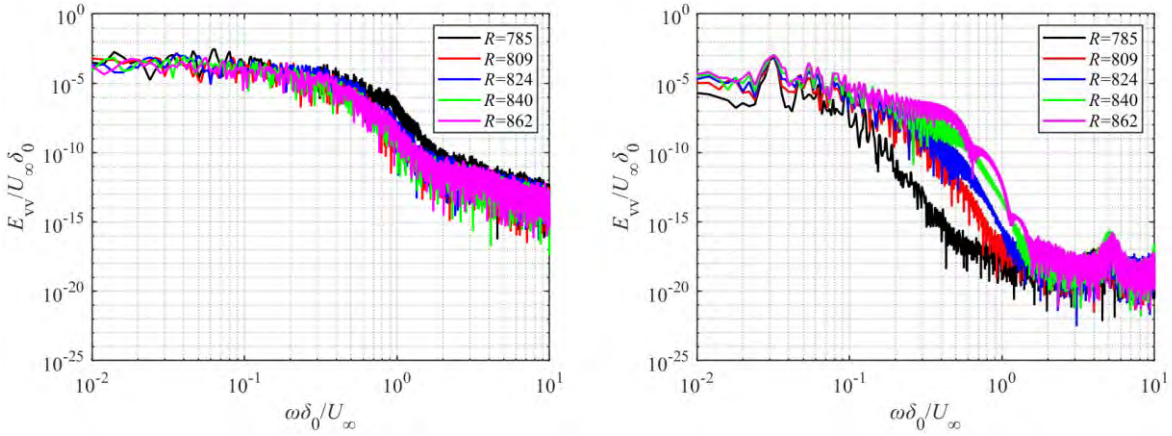
of normal velocity in the transition region gradually increase at y=0.1δ_0, which corresponds to the conclusion mentioned above that the fluctuations of the boundary layer displacement thickness intensify. The normal velocity at y=10δ_0 represents the monopole sound source of FW-H equation [19]. Dominant peaks of PSD curves in different regions also occur at $\omega\delta_0/U_\infty$=3.2×10^{-2}. Unlike the results of pulsating pressure, the normal velocity still increases in further downstream locations in the turbulence region. Therefore, the monopole noise generated in the turbulent region is theoretically much greater than that in the laminar flow region and the transition region.



(a) y=0.1δ_0 in laminar region **(b) y=10δ_0 in laminar region**



(c) y=0.1δ_0 in transition region **(d) y=10δ_0 in transition region**



(e) y=0.1δ_0 in turbulence region **(f) y=10δ_0 in turbulence region**

Fig. 10 PSD of normal velocity in different regions on the plate

Conclusions

In this paper, WRLES coupled with nonlinear PSE and porous-surface FW-H method is applied in the current simulation to resolve flow field and acoustic characteristics of the laminar-turbulent transition in a flat plate. Compared to the previous studies, our numerical results are convincing. The fluctuations of boundary layer displacement thickness, pressure, velocity and radiated noise in different regions of the flat plate are compared and analyzed. According to our simulation results, the disturbance modes imposed at the inlet boundary by PSE have a great influence on the fluctuations of pressure and normal velocity in the laminar region and early transition region. Spectral curves of these parameters have obvious peaks at the frequencies of TS wave and subharmonic wave especially in the near-wall region. From the onset of transition, the fluctuations of pressure, normal velocity and boundary layer displacement thickness increase significantly with the increase of R. In the late-transition region and fully developed turbulence region, the flow becomes more complex and is no longer dominated by the disturbance modes. Therefore, spectral for the above flow field parameters and radiated noise show a high order at the low frequency in the both transition and turbulence region without frequency peaks. Because the pulsating pressure and normal velocity gradually increase in the transition region, the radiated noise level in the transition region is slightly lower than that in the turbulence region.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (52131102), to which the authors are most grateful.

References

[1] Erhard, P., Etling, D., Muller, U., Riedel, U., Sreenivasan, K, R., (2010) Prandtl-essentials of fluid mechanics, Springer Science & Business Media
[2] Narasimha, R., (1985) The laminar-turbulent transition zone in the boundary layer, *Progress in Aerospace Sciences*, **22**(1): 29-80.
[3] Mayer, C, S, J., Von, Terzi, D, A., Fasel, H, F., (2011) Direct numerical simulation of complete transition to turbulence via oblique breakdown at Mach 3, *Journal of Fluid Mechanics*, **674**: 5-42.
[4] Subbareddy, P., and Candler, G. (2012) DNS of Transition To Turbulence in a Mach 6 Boundary Layer, 43rd AIAA Thermophysics Conference, New Orleans, Louisiana, American Institute of Aeronautics and Astronautics.
[5] Sayadi, T., Hamman, C, W., Moin, P., (2013) Direct numerical simulation of complete H-type and K-type transitions with implications for the dynamics of turbulent boundary layers, *Journal of Fluid Mechanics*, **724**: 480-509.
[6] Drikakis, D., Ritos, K., Spottswood, S, M., Riley, Z, B., (2021) Flow transition to turbulence and induced acoustics at Mach 6, *Physics of Fluids*, **33**(7).
[7] Mack, L, M., (1984) Boundary-layer linear stability theory, *Agard rep*, **709**(3): 1-3.
[8] Jee, S., Joo, J., Lin, R, S., (2018) Toward cost-effective boundary layer transition computations with large-eddy simulation, *Journal of Fluids Engineering*, **140**(11): 111201.
[9] Danvin, F., Olazabal-Loume, M., and Pinna, F., (2018) Laminar to turbulent transition prediction in hypersonic flows with metamodels., 2018 Fluid Dynamics Conference, Atlanta, Georgia, American Institute of Aeronautics and Astronautics.
[10] Bae, H., Lim, J., Kim, M., Jee, S., (2023) Direct-Numerical Simulation with the Stability Theory for Turbulent Transition in Hypersonic Boundary Layer, *International Journal of Aeronautical and Space Sciences*, **24**(4): 1004-1014.
[11] Herbert, T., (1997) Parabolized stability equations, *Annual Review of Fluid Mechanics*, **29**(1): 245-283.
[12] Lozano, D, A., Hack, M, J, P., Moin, P,. (2018) Modeling boundary-layer transition in direct and large-eddy simulations using parabolized stability equations, *Physical review fluids*, **3**(2): 023901.
[13] Kim, M., Lim, J., Kim, S., Jee, S., Park, J., Park, D., (2019) Large-eddy simulation with parabolized stability equations for turbulent transition using OpenFOAM, *Computers & Fluids*, **189**: 108-117.

[14] Olichevis, H, G, L., Fidkowski, K, J., Martins, J, R, R, A., (2021) Toward automatic parabolized stability equation-based transition-to-turbulence prediction for aerodynamic flows, *AIAA Journal*, **59**(2): 462-473.
[15] Smagorinsky, J., (1963) General circulation experiments with the primitive equations: I. The basic experiment, *Monthly weather review*, **91**(3): 99-164.
[16] Nicoud, F., and Ducros, F. (1999) Subgrid-Scale Stress Modelling Based on the Square of the Velocity Gradient Tensor, *Flow, Turbulence and Combustion*, **62**(3): 183-200.
[17] Herbert, T., Bertolotti, F, P., (1987) Stability analysis of nonparallel boundary layers, *Bull. Am. Phys. Soc*, **32**(2079): 590.
[18] Lighthill, M, J., (1952) On sound generated aerodynamically I. General theory, *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, **211**(1107): 564-587.
[19] Williams, J, E, F., Hawkings, D, L,. (1969) Sound generation by turbulence and surfaces in arbitrary motion, *Philosophical Transactions for the Royal Society of London. Series A, Mathematical and Physical Sciences*, **1969**: 321-342.
[20] Di Francescantonio, P., (1997) A new boundary integral formulation for the prediction of sound radiation, *Journal of Sound and Vibration*, **202**(4), 491–509.
[21] Brentner, K, S., Farassat, F,. (1998) Analytical comparison of the acoustic analogy and Kirchhoff formulation for moving surfaces, *AIAA journal*, **36**(8): 1379-1386.
[22] Epikhin, A., Evdokimov, I., Kraposhin, M., Kalugin, M., and Strijhak, S. (2015) Development of a Dynamic Library for Computational Aeroacoustics Applications Using the OpenFOAM Open Source Package, *Procedia Computer Science*, **66**: 150–157.
[23] Kachanov, Y, S., and Levchenko, V, Y., (1984) The resonant interaction of disturbances at laminar-turbulent transition in a boundary layer, Journal of Fluid Mechanics **138**, 209–247.
[24] Hunt, J, C, R., Wray, A, A., Moin, P,. (1988) Eddies, streams, and convergence zones in turbulent flows, Studying turbulence using numerical simulation databases, 2. Proceedings of the 1988 summer program.
[25] Emmons, H, W., (1951) The laminar-turbulent transition in a boundary layer-Part I, *Journal of the Aeronautical Sciences*, **18**(7): 490-498.
[26] Lauchle, G.C. (1981) Transition noise—The role of fluctuating displacement thickness, *The Journal of the Acoustical Society of America*, **69**(3), 665–671.

Subtype Classification of Circulating Rare Cells using Fluorescence-Topology Integrated Analysis for Cancer Prognosis

Yongjian Yu[1] and Jue Wang[2]

[1]Axon Connected, LLC, Earlysville, VA, USA
[2]Department of Mathematics, Union College, Schenectady, NY, USA
Corresponding author: wangj@union.edu

Abstract

Solid tumor metastasis involves the detachment of tumor cells into the vasculature or lymphatics. This study aims to improve the accuracy of phenotyping circulating rare cells (CRCs), which have been implicated in metastasis, tumor progression, and therapeutic resistance. Conventional deep learning object classification techniques rely on discriminative affine-transform invariant morphology and brightness features. However, their effectiveness is limited by highly occluded or distorted CRCs. Using four-channel immunofluorescence microscope data of lung and prostate cancers, we have developed a novel method for multi-subtype CRC classification. We propose a pioneering classification framework grounded in topologically invariant attributes and embedding geometrical shape via persistence homology. This method yields high accuracy, significantly surpassing conventional methods. This work demonstrates promising advancements in cancer management.

Keywords: Classification, circulating rare cells, fluorescence, topologically invariant, deep learning

1. Introduction

Liquid biopsy stands as a transformative approach in cancer management, offering a noninvasive avenue for multifaceted applications, including diagnosis, treatment selection, monitoring, and early recurrence detection. Immunofluorescence (IF) microscope assays rely on bespoke biomarkers, high-resolution imagery, and advanced cancer cell capture and classification models. This capability is achieved through combinations of specific antibodies tagged with fluorophores and is clinically relevant in oncology [7]. They serve as indispensable tools in both research and clinical realms. Additionally, their utilization extends to pharmaceutical companies, expediting the screening and development of promising new drug candidates and biomarkers.

The intricate process of solid tumor metastasis involves the detachment of tumor cells into the vasculature or lymphatics. Among these circulating entities, circulating tumor cells (CTCs) originate from epithelial cells of the primary tumor; circulating stromal cells (CSCs) derive from the tumor's connective tissue or supporting system; and circulating endothelial cells (CECs) arise from blood vessels supporting tumor growth. These circulating rare cells (CRCs) play crucial roles in cancer dynamics and have been implicated in metastasis, tumor progression, and therapeutic resistance [1].

Efficient classification of CRCs into clinically relevant subtypes can provide valuable prognostic insights. Apoptotic cells, exhibiting early signs of cellular breakdown, and anucleated cells, characterized by intact structure but devoid of nuclear material, have shown associations with treatment outcomes and progression-free survival. The presence of hybrid

cells, expressing both CRC and white blood cell markers, potentially signifies immune system activity.
In a visual object classification task, an analyzer identifies objects in images by recognizing their visible features based on learned knowledge of the objects' morphology and patterns. The challenge in analyzing multichannel immunofluorescence (IF) microscope data from the CRC assay lies in developing a comprehensive display algorithm that composes the data into a visible pseudo-color image, highlighting features necessary for subtle CRC subtyping with minimal confusion errors. Currently, when reading and interpreting the unstructured, raw image data, the analyzer needs to manipulate the controls of the multiple channel display interactively. These display controls include adjusting channel gains, selecting display windowing/leveling, and configuring the look-up table for each channel to reveal the expected features of CRC subtypes on a per-image basis. Consequently, analysis results are subject to high interobserver variability in CRC subtyping.

In this work, we present a topology-based CRC subtype classification method and demonstrate its performance and efficacy on composite CRC images. The rest of the paper is organized as follows: Section 2 discusses the motivation behind the proposed method and provides details of the analysis algorithms. Section 3 presents the method's performance and results, followed by a comparison with CNN-based composite image classification. Finally, we conclude in Section 4.

2. Methodology

Developing a robust, automatic CRC detector and subtype classifier based on blood smears remains a challenging task, as the target CRCs are very sparse and immersed in cluttered scenes. Each filter on the slide contains approximately 1 million white blood cells (WBCs). Cell touching and partial overlapping are inevitable. A quick scan of the slide with a low-resolution lens generates approximately 100 images, from which the CRC detection algorithm, based on the YoloV8m-p2 model (Ultralytics), identifies the locations of all CRC candidates with a sensitivity of 95%. Each candidate CRC is then imaged again using a high-resolution objective lens. If properly focused, the resultant high-resolution image captures the detailed biomechanical properties, which are then analyzed by the CRC subtype classifier.

Four clinically relevant subtypes are identified and defined as follows: the intact cell (InC), apoptotic cell (ApC), anucleated cell (AnC), and hybrid cell (HyB). Table 1 summarizes the subtypes of CRC fluorescence originating from the use of fluorophore-conjugated antibodies, which emit light upon excitation by shorter wavelength light. In our study, three wavelengths $\{\lambda_1, \lambda_2, \lambda_3\}$ are used, corresponding to three fluorescent dyes: DYLight@549 for cytokeratin (CK), DYLight@490 for pan-leukocyte cocktail (CD45), and 4′,6-diamidino-2- phenylindole (DAPI). Let I_{λ_i} denote the channel image corresponding to wavelength λ_i.

The characteristics of CRC subtypes are inadequately defined in terms of morphological and brightness patterns. It is observed that the CRC images annotated by experts do not exhibit affine transform similarity within the same subtype. Cultured cancer cells are mostly round with regular cellular and nuclear shapes, and the nucleus is typically centered and surrounded by cytokeratin. In contrast, CRCs may exhibit highly variable shapes (polymorphism), including round, oval, elongated, and clusters, as well as high distortion. This distortion represents cell stress imposed during sample preprocessing through immunocapture, structural incompleteness, nucleus-cytoplasm separation/disassociation typical of apoptosis, and partial occlusion in cluttered scenes. Sample images are shown in Figure 1.

Table 1. CTC subtype in terms of stain characterization.

Class ID	Subtype	I_{λ_1}	I_{λ_2}	I_{λ_3}
InC	Intact	CK+	CD45–	DAPI+
ApC	Apoptotic	CK~+	CD45–	DAPI~+
AnC	Anucleated	CK+	CD45–	DAPI–
HyB	Hybrid	CK+	CD45~+	DAPI+

+: positive, –: negative, ~+: mildly positive ; ~-: mildly negative

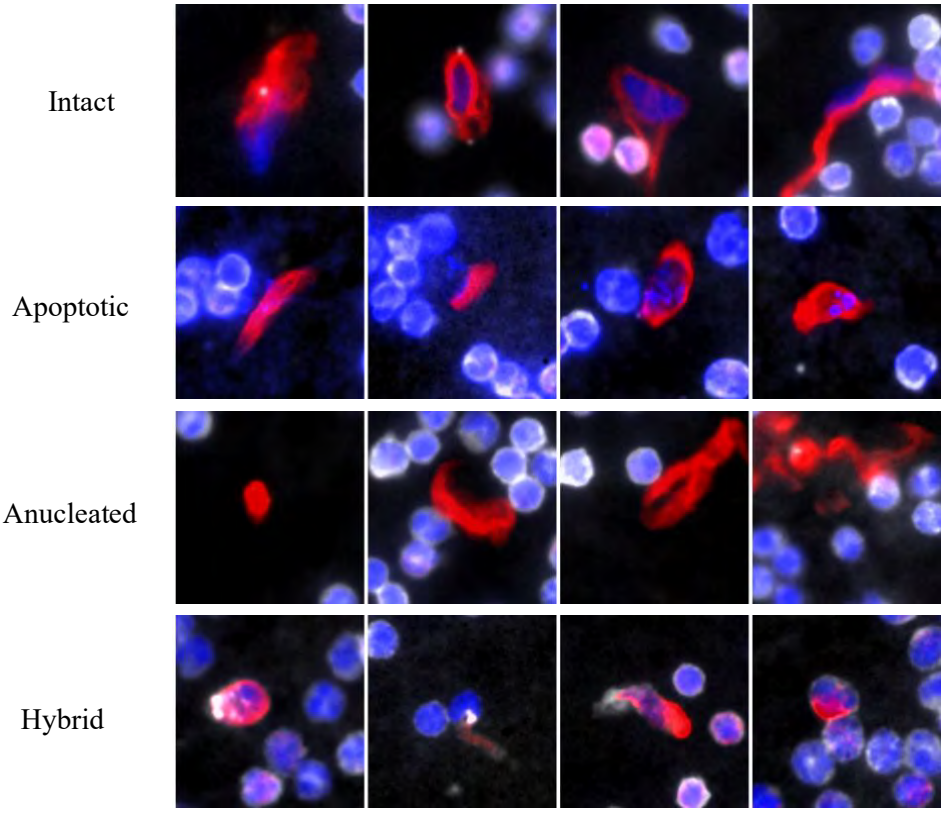


Figure 1. Example CRCs in each subtype showing a high irregularity in terms of morphology. For apoptotic CRCs, the blue channel gain is applied so heavily to show the DNA fragments that the nuclei of the WBCs overflow.

Conventional AI object classification/recognition techniques developed for computer vision applications exploit and rely heavily on the discriminative affine-transform invariant morphology and brightness features of objects. Their performance degrades significantly for occluded or distorted objects because learning affine-transform invariant class features is hardly feasible when the size of the CRC training dataset is limited.

To address the inadequacies of conventional techniques for CRC subtype stratification, we have developed a topology-based method to classify CRCs using measurable homological properties. Persistent homology (PH) [3, 5] is a primary technique in topological data analysis, based on the concept that topological features detected over varying scales are more likely to represent intrinsic features. All measurements are evaluated using segmentations of the cell structures in

the corresponding channels. The classifier takes these measured data and classifies the cells using the persistence homology method to qualitatively gauge shapes.

In topology, homology informally counts topological features such as connected components (0-dimensional homological features), holes (1-dimensional homological features), voids (2-dimensional homological features), and so forth. The counts of such k-dimensional holes are known as Betti numbers.

2.1 Classification of CTC subtypes

The CTCs are promising biomarkers for predicting tumor metastasis and patient survival prognosis. Current methods involve using the total CTC count to stratify local and metastatic tumors [1, 10]. The phenotypic heterogeneity of CTCs considers their dynamic differentiation between epithelial and mesenchymal states through a transitional hybrid state, which is valuable for the preoperative assessment of tumor metastasis and overall patient survival [9]. We aim to classify epithelial or stromal CTCs (labeled using immunofluorescence staining) into four subtypes: intact, apoptotic, anucleated, and hybrid, based on their staining patterns and intensities.

The initial CTC subtype definition utilizes only qualitative terms, as outlined in Table 1. The positivity or negativity of immunofluorescence expressions is determined visually under interactive display conditions. Forming a composite image from multiple channels for human or computer analysis has proven challenging. Classifying CTC subtypes with deep networks is inappropriate since these networks are designed to classify objects in pre-fixed images or frames.

We contribute two main advancements in this work. First, we utilize persistent homology to define the geometrical shape of CTCs, which are often morphologically challenging to define precisely. By applying persistent homology, we transform the notion of the CTC shape into a tractable form. In this topological paradigm, the shapes of CTC cytoplasm and nucleus are represented as persistent diagrams using the clouds of pixels expressing significant levels of CK and well-expressive DAPI pixels, respectively. A CTC is considered intact if its topology is isomorphic to a one-dimensional simplicial complex (a connected component) and a two-dimensional simplicial complex (representing holes). In contrast, the shape of apoptotic CTCs, as defined by CK-masked pixels, is isomorphic to at least two one-dimensional simplicial complexes and lacks any two-dimensional simplicial complex. The DAPI-masked pixels of intact CTCs exhibit no two-dimensional simplicial complexes.

Secondly, we establish statistical metrics for WBC expression in each channel to quantify the significance of corresponding stain expressions, enabling a metric-based assessment of positivity. We employ the Minimum Covariance Determinant (MCD) method [8], known for its speed and resistance to outliers, to estimate the means and variances of WBC fluorescent intensities in CK, nucleus, and CD45 channels, respectively. The segmentations are denoted Mc, Mn, and ∂M_w, respectively, where ∂M_w is the boundary of WBC segmentation M_w. Robust positivity/negativity assessments are rooted in the Mahalanobis distance between CTC fluorescence and WBC references.

Below, we outline the algorithm for classifying CTC subtypes based on cellular component fluorescent intensities and topological patterns corresponding to the outlined subtype characteristics.

Step 1: Preprocessing.

Given a high-resolution image in a domain Ω centered at a CTC position, we first crop an image patch with a domain Ω*'*, which is of a size covering an average sized CTC marginally. Then we segment all WBCs (see Section 2.2), resulting in a WBC segmentation M_w. Next, over the domain Ω*'* we perform suppression of scattering fluorescence from the WBCs (see Section 2.3).

Step 2: CTC stain positivity assessment with confidence.

To cope with the high variability in the data intensity, the strength of stain expression is assessed relatively against the corresponding WBC reference. We compute the WBC cross-talk CK statistics $s^{ck} = \left(s_1^{ck}, s_2^{ck}\right)$, a vector of trimmed mean and standard deviation, and the *maximum outlier distances* d_{max}^{ck} using all pixels over $M_w \backslash \Omega'$ and the MCD method. A pixel at position p is declared to be CK marker positive if its outlier distance $d(p, s^{ck})$ satisfies $S(d(p, s^{ck}) - d_{max}^{ck}) > 0.5$ and $I_{ck}(p) > (s_1^{ck} + s_2^{ck})$. Denote the sigmoid function $S(x)$. The region Ω*'* is declared to be *CK positive* if the cardinality of the set of *CK positive* pixels is greater than a preset cutoff equal to one half of the average CTC cytoplasm area.

Assuming that the intensity of the CD45 marker expression follows a normal distribution and using the *maximum likelihood* estimation method, we calculate the WBC's CD45 marker statistics $s^{wc} = (s_1^{wc}, s_2^{wc})$ using the pixels in $M_w \backslash \Omega'$. Likewise, we compute the background statistics $\{b^c = (b_1^c, b_2^c)\}_{c=1,2,3}$ using the pixels in $(\sim M_w) \bigcap (\Omega \backslash \Omega')$ in case that the scene contains no WBCs. A pixel at position $p \in \Omega'$ is declared to be positive to CD45 marker if its intensity satisfies $I_W(p) > \max\{(s_1^w - s_2^w), (b_1^w + 3b_2^w)\}$; the point is declared to be moderately positive if $(s_1^w - 2s_2^w) \leq I_W(p) < (s_1^w - s_2^w)$, or the point p is considered CD45 maker negative if $I_W(p) < (s_1^w - 3s_2^w)$.

The region Ω*'* is declared to be WBC negative if the count of CD45 positive pixels is less than one half of the average WBC perimeter; it is declared to be WBC moderately positive if the count of moderately positive pixels is greater than one half of the average WBC perimeter.

A pixel at position $p \in \Omega'$ is declared to be DAPI positive if $I_C(p) > \max\{(s_1^N - 3s_2^N), (b_1^N + 3b_2^N)\}$. The region Ω*'* is declared to be Nucleus stain positive if the count of DAPI positive pixels is greater than one half of average WBC nucleus size. Otherwise, it is declared to be free of nuclear (DNA) materials.

Step 3: CTC shape via persistence analysis.

If the region Ω' is neither positive nor moderately positive to WBC, the following is performed.

a) Collect CK well-stained pixels $\left\{p_1^{ck}, p_2^{ck}, \cdots, p_{n_c}^{ck}\right\}$ in $\Omega' \backslash M_w$ for evaluating the CTC CK persistence, where each point p_1^{ck} is a vector of its (x, y) coordinates, n_c is the number of points, d_{max}^{ck} is the maximal distance of the WBC outliers in the CK channel, and k is a weight factor.

b) Placing the growing disks around each CK positive pixel and keeping track of when they intersect, we quantify when the topological features such as connected components (0-dimensional persistent homology) and loops (1-dimensional persistent homology) are "born" or "die".

c) The significant CK persistence provides topological shape generators H_0(CK) and H_1(CK) for the differentiation of intact and apoptotic CTCs in terms of CK stain shape.
d) Similarly, we collect a cloud of DAPI well-stained pixels $\{p_1^N, p_2^N, \cdots, p_{n_N}^N\}$ in $\Omega' \backslash M_w$ and derive a topological shape generator H_0(DAPI) for the differentiation of intact and apoptotic CTCs in terms of nuclear stain shape.

If the region Ω' is WBC positive, the algorithm execution is terminated and the CTC candidate is assigned to be debris.

Step 4: Combinatorial condition subtype.
With the positivity information and shape information collected, it is then straightforward to infer the CTC subtypes according to the prescribed characteristics listed in Table 2. If an entry is not classifiable using the proposed subtype classifier, it is declared to be unknown.

Table 2. CTC subtype classification in terms of persistent homology and fluorescence positivity.

ID	Subtype	CK channel	WBC channel	nuclear channel
InC	Intact	CK+, H_1=0	CD45–	DAPI+, H_0=1
ApC	Apoptotic	CK~+, H_1>1	CD45–	DAPI~+
		CK~+		DAPI~+, H_0>1
AnC	Anucleated	CK+	CD45–	DAPI–
HyB	Hybrid	CK+	CD45~+	DAPI+

2.2 WBC Segmentation

We now describe the method for segmenting WBCs into a binary mask M_w. WBCs are identified by their expression of the pan-leukocyte cocktail CD45 (CD45+), which marks the cell membrane. The WBC channel I_{λ_2} (or I_w) presents challenges due to its low contrast and high noise, making robust WBC segmentation difficult. Conversely, the DAPI (4′,6-diamidino-2-phenylindole) fluorescence of nuclear material in channel I_{λ_3} (or I_N) offers high quality, high contrast, and minimal noise. Therefore, an effective approach is to initially segment the nuclear (DNA) material predominantly within the WBCs, and subsequently use this nuclear segmentation as seeds for segmenting the WBCs themselves.

2.2.1 WBC nuclear segmentation algorithm

The nuclei of WBCs exhibit distinct structures across various scales. The most common type, lymphocytes, typically feature a smaller nucleus-to-cytoplasm ratio and numerous granules. Lymphocytes vary widely in size, with larger cells measuring 10–14 μm and smaller ones ranging from 6–9 μm [6].

Monocytes, on the other hand, possess a single, large nucleus that often displays folds and is typically kidney-shaped, measuring approximately 15–22 μm in diameter. Given these characteristics of Peripheral Blood Mononuclear Cells (PBMCs), we employ segmentation of WBC nuclei in the wavelet transform domain, enabling multi-scale views of the objects of interest. By selecting appropriate scales that match WBC nuclear sizes, we apply a locally adaptive thresholding technique to achieve nuclear segmentation. This process is followed by morphological cleaning and a two-stage procedure to split nuclear clusters. The segmentation is illustrated in Figure 2. The algorithm is outlined as follows:

Step 1. Multiscale adaptive thresholding

1) Apply a four-level À Trous discrete wavelet transform equipped with a B_3-spline scaling function to the nuclear channel image I_N (or I_{λ_2}), resulting in four detailed images W{1}, W{2}, W{3}, W{4}, where W{k} is the scale image of k pixels, and a smoothed image W{5} all with the same size as the original image I_N.
2) Threshold I_N at W{2}+W{5} followed by a morphological opening with a circular structural element of radius 3 and a hole filling, resulting in a binary mask B.

Step 2. Mask cleaning

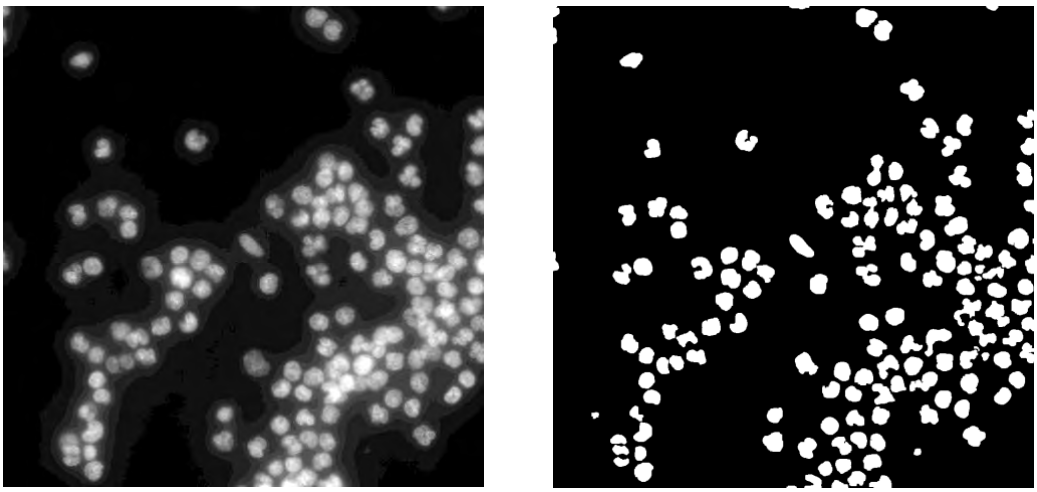
Apply a morphological erosion to B with a circular structural element of radius 5, resulting in B_1.

Step 3. Cluster splitting stage 1

1) Label the components in B_1 then apply image morphological dilation, resulting in a labeled image L_1.
2) Find the edges in L_1 and subtract them from B and filter out small blobs less than 50 pixels gives the segmentation of nuclear materials, denoted by B_2.

Step 4. Cluster splitting stage 2

1) Check if there is any remaining cluster of nuclei and split them if existed.
2) Extract cluster blobs in B_2 that are larger than a pre-determined upper cutoff of nucleus size, giving B_{2C}. If B_{2C} is empty, let $B_N = B_2$ and go to 8). Otherwise, proceed to 3).
3) Invert the input image I, resulting in I'.
4) Apply adaptive mean thresholding with rotational line structural elements of length 91 and a sensitivity of 0.5 to I' yielding M.
5) Subtract M from B_{2C} yielding B_{2CS}.
6) Find elements of B_{2CS} that touch B_{2C} followed by boundary smoothing using morphological erosion with circular structural element of radius 1, yielding B_{2CSS}.
7) Compute the union (B_2 - B_{2C}) and B_{2CSS} followed by removal of tiny blobs yielding nuclear instance segmentation B_N.
8) B_N is the final nuclear material mask including both the nuclei of WBCs and those to be associated with CRCs.



(a) I_N (b) B_N

Figure 2. (a) Nucleus channel image. (b) DNA instance segmentation B_N where nuclear clusters are split into individual nucleus (of monocyte, lymphocyte) and nuclear lobes (of neutrophil).

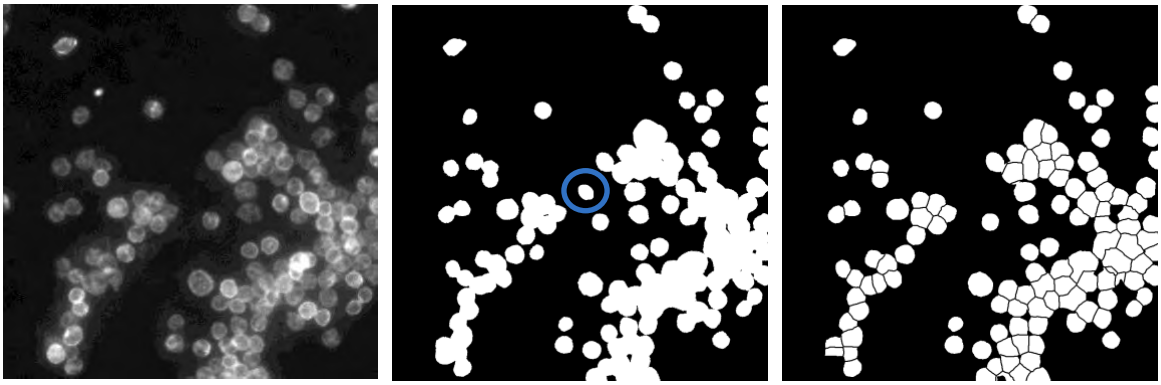
Debris blobs are eliminated using a two-iteration 3x3 erosion particle filter. Cell and nuclear sizes are determined by counting the number of above-threshold pixels in the cytokeratin (CK) and DAPI channels, respectively. The results of the cell and nuclear size calculations undergo a filtering step to exclude cells with implausible nuclear sizes, defined as cases where the nuclear area exceeds 95% of the cell area.

2.2.2 WBC segmentation algorithm

For WBC segmentation, we employ the filtered B_N as seeds after eliminating unwanted blobs, which may include broken nuclei or debris. The segmentation is conducted using a region-based active contour method. The segmentation is illustrated in Figure 3. The algorithm is outlined as follows:

Step 1. Initial segmentation

1) The unwanted blobs in B_N are those having areas less than 50 pixels or greater than 3000 pixels and a ratio of minor to major axis length less than 0.5. The filter output is denoted by B_{NF}.
2) For each blob in B_{NF}, a circular mask is fit that is centered at the blob centroid with a radius equal to one half of the major axis length. The resultant circular masks represent the initial segmentation of the WBC.
3) The initial segmentation contours only approximate the WBC boundaries. To refine the initial segmentation using the WBC channel image is suboptimal as the WBC channel is of low contrast and high noise. We use a weighted sum of the nuclear channel and WBC channel to guide the refinement. The weights are selected such that the mean foreground signals in the two channels are equal.
4) Remove non-WBC areas from the WBC mask by removing the blobs with a low WBC marker level compared to the background WBC mean fluorescence intensity (MFI), defined as $T := M\{I_W, \sim B_W\} + k \cdot S\{I_W, \sim B_W\}$, where $M\{\}$ and $S\{\}$ stand for mean and standard deviation operations, respectively; $\sim B_W$ denotes the complement of B_W; k is a positive factor greater than 1 and less than 3.



(a) I_W (b) B_W (c) B'_W

Figure 3. (a) WBC channel image (Intact). (b) seeded WBC segmentation B_W including non-WBC (indicated by the blue circle) due to some seeded blobs being either CRCs or other non-WBC DNAs. (c) Instance segmentation B'_W of WBCs through splitting with nucleus instance segmentation's watershed lines and non-WBC removal.

2.3 Scatter reduction preprocessing

The light scattering caused by WBCs increases the background autofluorescence intensity in the blue and white channels, which can interfere with the topological analysis of CTCs when they are in proximity to WBCs. Therefore, it is crucial to apply a scatter reduction operation to the data to accurately reconstruct the underlying CTC topology for subtype identification.

We model the scattering effect of WBCs at a point p within the CTC region using a negative exponential function of the distance between the point and the nearest WBC region boundary. The scattering effect diminishes with increasing distance from a WBC. The scatter-suppressed data at point p is approximated by multiplying the original data $I(p)$ with a suppression function defined as:

$$f(x) = \begin{cases} e^{1-\frac{x}{d}}, & x < d \\ 1, & \text{else} \end{cases}$$

where x is the distance between p and the nearest WBC in the region. The d is a positive parameter; its value is channel dependent. Since the WBC does not stain red, $f(x) \equiv 1$ for the red channel data. For the blue and white channels, the value of d ranges from 10 to 20 pixels determined experimentally.

3. Experiments and Results

3.1 Data preparation

The CTC/CSC testing procedure begins with separating blood components from the whole blood sample using centrifugation and filtration [2, 4]. Following separation, PBMCs — including lymphocytes, monocytes, CTCs, and CSCs — contained in the buffy coat are fixed, permeabilized, and stained using two distinct reagent packages: one for the CTC test and another for the CSC test. Subsequently, the cells are deposited onto membranes, mounted on glass slides, and scanned using a fluorescence microscope to detect and categorize CK+CD45-DAPI+ CTCs/CSCs. High-resolution images captured with a 40X objective for subtype identification are 1024-by-1024 pixels in size and have a 16-bit depth. To construct the ground truth dataset, multiple certified technicians positively identify and label the CTCs/CSCs based on size, shape, and fluorescence intensity of each candidate cell, resulting in an annotated training dataset comprising 1,320 CRCs.

The nCyte microscope operates as a four-channel system, where channels 1 to 3 are designated for cancer markers, white blood cell markers, and nucleus markers, respectively. Channel 4 is reserved for custom markers indicative of specific cancer types or potential drug targets. For example, CA-125 serves as a custom marker for ovarian cancer, while HER2 indicates a target for immunotherapy.

In our study, whole blood samples from patients with prostate cancer and small cell lung cancer (SCLC) are utilized. High-resolution images with a resolution of 0.16 μm/pixel are captured at locations where CK+DAPI+CD45- events are identified through low-resolution scanning of the slide and subsequent detection processing using YOLOv3 as the core method.

3.2 Topological classification

We selected the VGG16 classifier as our baseline for comparison. However, since VGG16 is pretrained using 8-bit RGB images and does not support 16-bit per channel data, we developed visualization algorithms to convert the 4-channel, 16-bit images from the digital microscope

into 8-bit RGB images using histogram-based techniques. This conversion ensures that the output images have pixel values scaled appropriately to accurately display the morphology of intact CRCs, including nucleus, cytoplasm, and membrane structures. The post-processed CRC images are illustrated in Figure 4.

Our CRC classification achieved an overall accuracy of 87.54% on our training and validation dataset, surpassing the 62% accuracy obtained using conventional multi-class CNN classifiers. Detailed confusion matrices are provided in Tables 3 and 4. Future improvements are anticipated through the implementation of cancer-type specific scan profiles, optimized AI hyperparameters, and the use of higher quality data with meticulously verified and consistent annotations.

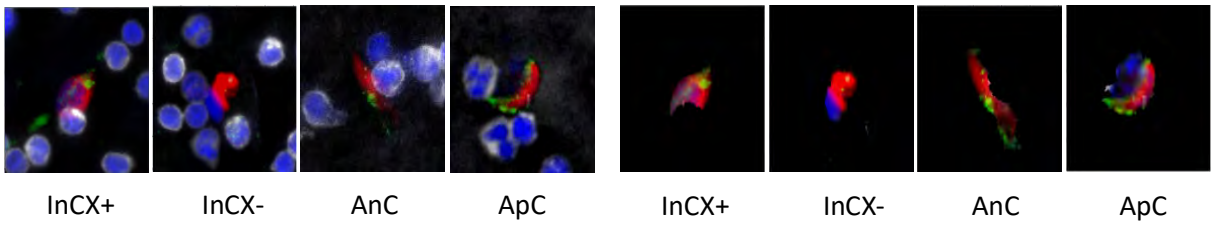


Figure 4. Left: lung cancer CSCs. Right: with removal of WBCs.

Table 3. CNN classifier confusion matrix for CSC subtype classification.

	Predicted subtype					
True subtype		InC	ApC	AnC	HyB	Recall
	InC	143	10	1	17	83.63%
	ApC	41	66	5	5	56.41%
	AnC	54	9	99	0	61.11%
	HyB	75	9	1	60	41.38%
	Precision	45.69%	93.4%	93.4%	73.17%	

Table 4. Topologically-invariant classifier confusion matrix for CSC subtype classification.

	Predicted subtype					
True subtype		InC	ApC	AnC	HyB	Recall
	InC	118	0	0	2	98.33%
	ApC	11	39	9	2	63.93%
	AnC	11	0	89	2	87.25%
	HyB	1	0	0	21	95.45%
	Precision	83.69%	100%	90.82%	77.78%	

4. Conclusions

To address the challenge of subtype classification of circulating rare cells (CRCs), we have developed a fluorescence-topology combinatorial condition classifier. This approach leverages in-image learned knowledge to quantify cell expression and incorporates shape persistence homology encoding. Our novel method demonstrates robustness and resilience against CRC distortion and occlusion, achieving a classification accuracy of 88% for CRC subtypes that carry clinical significance.

Acknowledgments

This work was supported by National Science Foundation (NSF) SBIR Phase II Project "SBIR Phase II: Development of AI Software to Capture and Identify Circulating Rare Cells in Lung Patients" Award Number: 2230782.

References

[1] J.S. Ankeny, C.M. Court and S. Hou, "Circulating tumor cells as a biomarker for diagnosis and staging in pancreatic cancer", Br J Cancer, 114(12), 1367–1375, 2016.

[2] F.A. Coumans, G. van Dalum, M. Beck and L. W. Terstappen, "Filter characteristics influencing circulating tumor cell enrichment from whole blood", PLoS One 8: e61770, 2013.

[3] P. Edwards P, K. Skruber, N. Milićević, J.B., Heidings, et al., "TDAExplore: Quantitative analysis of fluorescence microscopy images through topology-based machine learning", Patterns, 2(11), 100367, 2021.

[4] T. Fehm, E.F., Solomayer, S. Meng, T. Tucker, et al., "Methods for isolating circulating epithelial cells and criteria for their classification as carcinoma cells", Cytotherapy, 7(2), 171–185, 2005.

[5] A.A. Malek, M.A. Alias, F.A. Razak, M.S.M. Noorani, et al., "Persistent homology-based machine learning method for filtering and classifying mammographic microcalcification images in early cancer detection", Cancers (Basel), 15(9): 2606, 2023.

[6] S. Meng, D. Tripathy, E.P. Frenkel, S. Shete, et al., "Circulating tumor cells in patients with breast cancer dormancy", Clin Cancer Res, 10(24), 8152–8162, 2004.

[7] J.Y. Pierga, C. Bonneton, A. Vincent-Salomon, P. de Cremoux, et al., "Clinical significance of immunocytochemical detection of tumor cells using digital microscopy in peripheral blood and bone marrow of breast cancer patients", Clin Cancer Res, 10(4), 1392–1400, 2004.

[8] P.J. Rousseeuw and A.M. Leroy, "Robust regression and outlier detection", John Wiley & Sons, Inc., 1987.

[9] Y. Sun, G. Wu, K.S. Cheng, A. Chen, et al., "CTC phenotyping for a preoperative assessment of tumor metastasis and overall survival of pancreatic ductal adenocarcinoma patients," eBioMedicine, 46: 133–149 2019.

[10] Y.W. Tien, H.C. Kuo and B.I. Ho, "A high circulating tumor cell count in portal vein predicts liver metastasis from periampullary or pancreatic cancer: a high portal venous CTC count predicts liver metastases", Medicine (Baltimore), 95(16): e3407, 2016.

Numerical Simulation of the Flow around Two Square Cylinders by MPS Method

Congyi Huang[1*], Jifei Wang[2], Decheng Wan[1†]

[1] Computational Marine Hydrodynamic Lab (CMHL), School of Naval Architecture, Ocean and Civil Engineering, Shanghai Jiao Tong University, Shanghai, China
[2] Shanghai Aerospace System Engineering Institute, Shanghai, China

*Presenting author: cyhuang@sjtu.edu.cn
†Corresponding author: dcwan@sjtu.edu.cn

Abstract

The problem of flow around columns is one of the most classic fluid problems. In engineering practice, the column often does not exist alone, but multiple cylinders exist simultaneously as a group. Thus, simulation of the flow around two columns is closer to the actuality. The fluid force act on the cylinder is affected by the interference between the cylinders, so the mechanism of the flow around the cylinders is more complicated. In this paper, the MLParticle-SJTU solver developed based on the MPS method (Moving Particle Semi-implicate Method) is used to simulate the flow around the two square columns in a row in the direction of the fluid, and the influence of the distance between the two columns on the flow is studied. Firstly, the Poiseuille flow is simulated with the MLParticle-SJTU solver and the obtained result is compared with the analytical solution to verify the reliability and accuracy of the solver. Then, the flow around the two columnsis simulated, and the influence of the spacing is analyzed from the flow field, vorticity field and lift-drag coefficient.

Keywords: MPS method, flow past square cylinders, MLParticle-SJTU solver.

Introduction

Flow past two columns in a row is a very typical problem, which is very common in engineering, and many scholars have carried out research on this problem. Sakamoto et al.(1987) studied the influence of the spacing between the two columns in tandem on the force of the column through experiments[1]. Moon Kyoung Kim et al.(2008) studied the velocity, turbulence intensity and Reynolds shear stress of the flow field around two square columns arranged in a row[2]. It was found that when Reynolds number is 5300 and 16000, the flow pattern at the spacing ratio H/D$\leqslant$2.5 is completely different from that at the spacing ratio H/D$\geqslant$3. Sohankar (2009)(2012) used FVM to conduct numerical simulation of flow pass two square cylinders at different Reynolds numbers[3][4]. The influence of the spacing ratio on the flow characteristics under different Reynolds numbers was analyzed. According to the value of the spacing ratio, the flow around the columns can be divided into three main states: single slender body flow field, reattachment flow field and co-shedding flow field. In 2012, Dipankar Chatterjee et al. (2012) used FLUENT to study the overall flow and heat transfer characteristics of the flow around the column under different Reynolds numbers and spacing ratios, such as total resistance coefficient and lift coefficient, local and surface average Nusselt number and Strouhal number [5]. Izadpanah Ehsan et al. (2012) systematically studied two-way column flow in power-law fluid for the first time [6]. It is found that with the increase of the power law exponent, the Reynolds number increases when the flow separation begins. Deepak Kumar (2017) used the FEM to study the flow around a tandem square

column with a Reynolds number of 40, and found that with the increase of Reynolds number, there are four kinds of flow patterns [7].

Numerical method

In this paper, a meshless particle method, the Moving Particle Semi-implicit Method (MPS) is used to simulate the liquid sloshing inside a spherical liquid cabin. The MPS method is a CFD method based on the Lagrangian representation. Different from the traditional mesh-based methods, the MPS method discretizes the computational domain into a series of particles, which do not have a fixed topological relationship with each other but carry their own physical information, such as displacements, velocities, accelerations, etc., respectively. Compared with the traditional mesh-based methods, the particle methods has the advantage that, since there is no mesh in its computation, there will be no mesh distortion or brokenness when simulating the violent flows with free surface large deformation. And the meshless particle method has high quality in tracking the free surface, dealing with the moving boundary and other problems. Therefore, the meshless particle method is very suitable for simulating problems with free surface large deformation characteristics, such as dam-break, water entry and liquid sloshing.

A lot of works based on the MPS method has been carried out by our group, including algorithm improvement, solver development, and applications. For solving the pressure oscillation problems that often occur in the traditional MPS method when calculating the pressure by the Poisson pressure equations, the group has proposed a set of improved MPS algorithms by improving the pressure solving method, the pressure gradient model, and the method of judging the free surface [8]. Based on this improved MPS method, our group has developed a meshless particle method solver MLParticle-SJTU, which can be run on the CPU, and the multi-CPU parallel computing can also be realized [9]. In order to make full use of the advantages of the MPS method in simulating violent flows, the group also developed algorithms for coupling the MPS method with other methods, including the MPS-FEM coupling algorithm for simulating fluid-structure coupling phenomena and independently developed the MPSFEM-SJTU solver [10]. And the MPSDEM coupling algorithm for simulating particle flows and independently developed the MPSDEM-SJTU solver [11]. However, due to the low efficiency, the MPS method cannot be used in the simulation of large-scale large scale computational applications. Therefore, a high-performance meshless particle solver MPSGPU-SJTU which can run on GPU is developed by our group, and it can improve the computational speed greatly while guaranteeing the precision and accuracy of the original particle solver [12]. In previous published papers, the numerical method of the MPS method has been introduced in detail [13-14]. Therefore, in this paper, the MPS method will be introduced just simply in the following part.

Governing equations

The governing equations for MPS method is shown as Eqs. (1), (2).

$$\frac{1}{\rho}\frac{\mathrm{D}\rho}{\mathrm{D}t} = -\nabla \cdot \boldsymbol{V} = 0 \tag{1}$$

$$\frac{\mathrm{D}\boldsymbol{V}}{\mathrm{D}t} = -\frac{1}{\rho}\nabla p + \nu\nabla^2\boldsymbol{V} + \boldsymbol{g} \tag{2}$$

where ρ is the fluid density, $\boldsymbol{V}$ is the velocity vector, p is the pressure, ν is kinematic viscosity, $\boldsymbol{g}$ is gravitational acceleration vector and t indicates time.

Particle interaction models

In MPS, particles interact with each other by a kernel function. The kernel function used in this paper is given in Eqs. (3a), 3(b).

$$W(r)=\frac{r_e}{0.85r+0.15r_e}-1,\ 0\le r<r_e \tag{3a}$$

$$W(r)=0,\ r_e\le r \tag{3a}$$

where r is the particle distance, r_e is the influence radius

The following equations are the particle interaction models used to discretize the terms in the governing equations.

$$\langle\nabla P\rangle_i=\frac{D}{n^0}\sum_{j\neq i}\frac{P_j+P_i}{\left|\boldsymbol{r}_j-\boldsymbol{r}_i\right|^2}(\boldsymbol{r}_j-\boldsymbol{r}_i)W(\left|\boldsymbol{r}_j-\boldsymbol{r}_i\right|) \tag{4}$$

$$\langle\nabla\cdot\boldsymbol{V}\rangle_i=\frac{D}{n^0}\sum_{j\neq i}\frac{(\boldsymbol{V}_j-\boldsymbol{V}_i)\cdot(\boldsymbol{r}_j-\boldsymbol{r}_i)}{\left|\boldsymbol{r}_j-\boldsymbol{r}_i\right|^2}W(\left|\boldsymbol{r}_j-\boldsymbol{r}_i\right|) \tag{5}$$

$$\langle\nabla^2\phi\rangle_i=\frac{2D}{n^0\lambda}\sum_{j\neq i}(\phi_j-\phi_i)W(\left|\boldsymbol{r}_j-\boldsymbol{r}_i\right|) \tag{6}$$

$$\lambda=\frac{\sum_{j\neq i}W(\left|\boldsymbol{r}_j-\boldsymbol{r}_i\right|)\left|\boldsymbol{r}_j-\boldsymbol{r}_i\right|^2}{\sum_{j\neq i}W(\left|\boldsymbol{r}_j-\boldsymbol{r}_i\right|)} \tag{7}$$

$$\langle n\rangle_i=\sum_{j\neq i}W(\left|\boldsymbol{r}_j-\boldsymbol{r}_i\right|) \tag{8}$$

where D is the number of dimensions, $\boldsymbol{r}$ is the position vector and n^0 is the particle number density at the initial time, and the particle number density is defined as Eq. (8).

Pressure Poisson equation

In this paper, the pressure is calculated by solving the mixed source term Pressure Poisson equation (PPE) proposed by Tanaka and Masunaga[15], Lee et al.[16], which can be written as the following equation.

$$\langle\nabla^2 p^{k+1}\rangle_i=(1-\gamma)\frac{\rho}{\Delta t}\nabla\cdot\boldsymbol{V}_i^*-\gamma\frac{\rho}{\Delta t^2}\frac{\langle n^k\rangle_i-n^0}{n^0} \tag{9}$$

where p^{k+1}, Δt and $\boldsymbol{V}_i^*$ are the pressure of the step $k+1$, time step and temporal velocity and γ is the weight of the particle number density term between 0 to 1.

Free surface detection

The particle number density is used to determine whether a particle is a free surface particle. If $\langle n\rangle_i/n^0<0.80$, the particle will be determined as a free surface particle. If $\langle n\rangle_i/n^0>0.97$, the particle will be regarded as an internal fluid particle. If $0.80<\langle n\rangle_i/n^0<0.97$, an additional function needs to be introduced.

$$\langle \boldsymbol{F} \rangle_i = \frac{D}{n^0} \frac{(\boldsymbol{r}_i - \boldsymbol{r}_j)}{|\boldsymbol{r}_i - \boldsymbol{r}_j|} \sum_{j \neq i} W(|\boldsymbol{r}_i - \boldsymbol{r}_j|) \tag{10}$$

where $\langle \boldsymbol{F} \rangle_i$ can be regarded as the asymmetric arrangements of neighbor particles. $|\boldsymbol{F}|^0$ is the initial value of $|\boldsymbol{F}|$ for surface particles. If $\langle |\boldsymbol{F}| \rangle_i > 0.9|\boldsymbol{F}|^0$, the particle will be regarded as a free surface particle.

Boundary condition

The multilayer particles boundary condition is applied to fulfil the support region of the particles near the boundary, as shown in Fig. 1. The pressure of the wall particle is calculated by solving the PPE equation, and ghost particle pressure is obtained by interpolation.

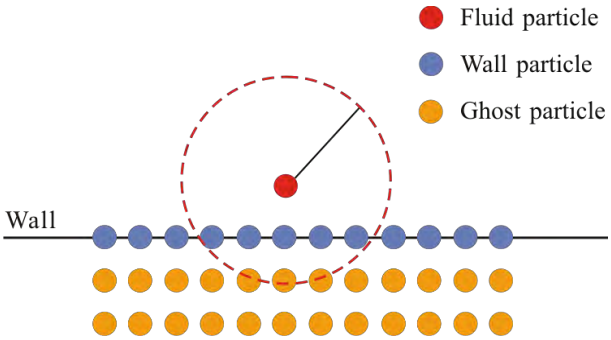


Fig. 1 Schematic of boundary particles

Free surface boundary condition

Virtual particles are arranged over the free surface to fulfill the support domain of the free surface particle, as shown in Fig. 2. And the particle number density and the pressure Laplacian model is modified as:

$$\langle n \rangle_i = \sum_{j \neq i} W_{ji} + \sum_{\text{virtual}} W_{\text{virtual},i} = + \sum_{\text{virtual}} W_{\text{virtual},i} \tag{11}$$

$$\left\langle \nabla^2 p^{k+1} \right\rangle_i = \frac{2D}{n^0 \lambda} \sum_{j \neq i} (p_j - p_i) W_{ji} + \frac{2D}{n^0 \lambda} \sum_{\text{virtual}} (p_{\text{virtual}} - p_i) W_{\text{vritual},i} \tag{12}$$

where n_i^* and $W_{\text{vritual},i}$ denote the particle number density of actual neighboring particles and virtual neighboring particles separately.

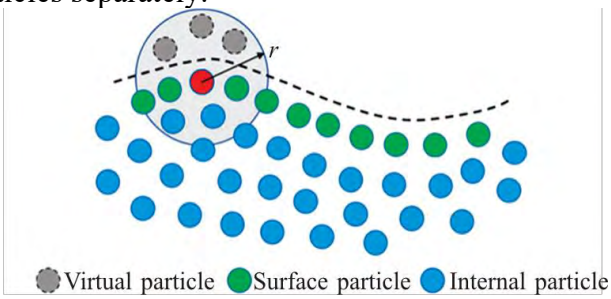


Fig. 2 Schematic of free surface boundary condition

Surface tension model

The surface tension is imposed by adding a source term to the governing equations, which is proposed by Brackbill et al.[17], as the following equations show.

$$\rho \frac{\mathrm{D}\boldsymbol{V}}{\mathrm{D}t} = -\nabla p + \mu \nabla^2 \boldsymbol{V} + \boldsymbol{F}^V + \boldsymbol{F}^S \tag{13}$$

$$\boldsymbol{F}^S = \sigma \kappa \nabla C \tag{14}$$

where σ is surface tension coefficient, κ is inter- face curvature and C is function to mark the free surface particles. For free surface particles $C = 1$, and for internal particles $C = 0$.

Numerical Simulation

Validation of the MLParticle-SJTU solver

Firstly, the effectiveness and accuracy of the MLParticle-SJTU solver are verified by simulating Poiseuille flow. The analytical formula of Poiseuille flow can be written as Eq.11.

(15)

The pipe diameter is h = 0.2m, the pipe length is s 15h, and the fluid density is ρ=1000 kg/m3, viscosity coefficient μ=0.001pa·s, the speed is 0.00002m/s, and the Reynolds number is 4. The velocity profile of the velocity in the x-direction at the section where x=10h is shown in Figure 3. As the flow develops, the flow velocity at the central line of the pipe increases continuously. When the dimensionless time Ut/h=0.5, the flow has developed to a steady state, and the velocity profile is parabolic, which almost coincides with the analytical solution. When Ut/h=10, the peak velocity of the section is slightly smaller than the analytical solution, which may be due to the small oscillation of the velocity field. This can verify the validity and accuracy of the MLParticle-SJTU solver.

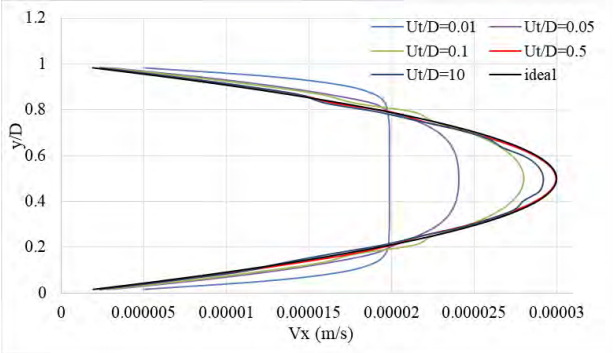


Fig. 3 Velocity profile of the cross section at x=10h

Simulation of the flow pass two square columns in tandem

In this section, the MLParticle solver is used to simulate flow pass two square columns in tandem. The computational domain scale is shown as Figure 4. The origin of the calculation coordinate system is at the center of the upstream square column, and the positive direction of the x-axis is the direction of the incoming flow. H is the distance between the centers of the two square columns, H/D=2.0~6.0. The particle spacing is set to D/Δx=20.

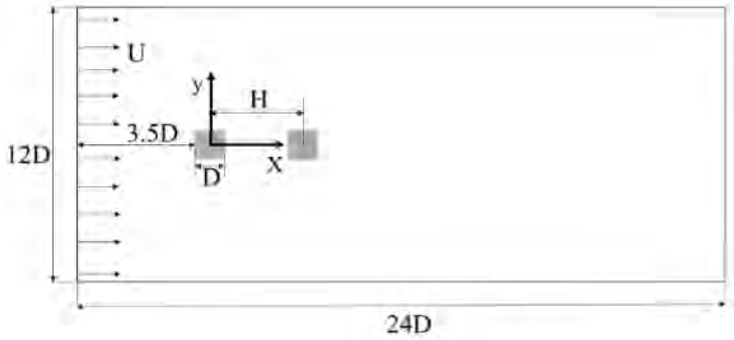


Fig. 4 Computational domain of flow pass two square cylinders

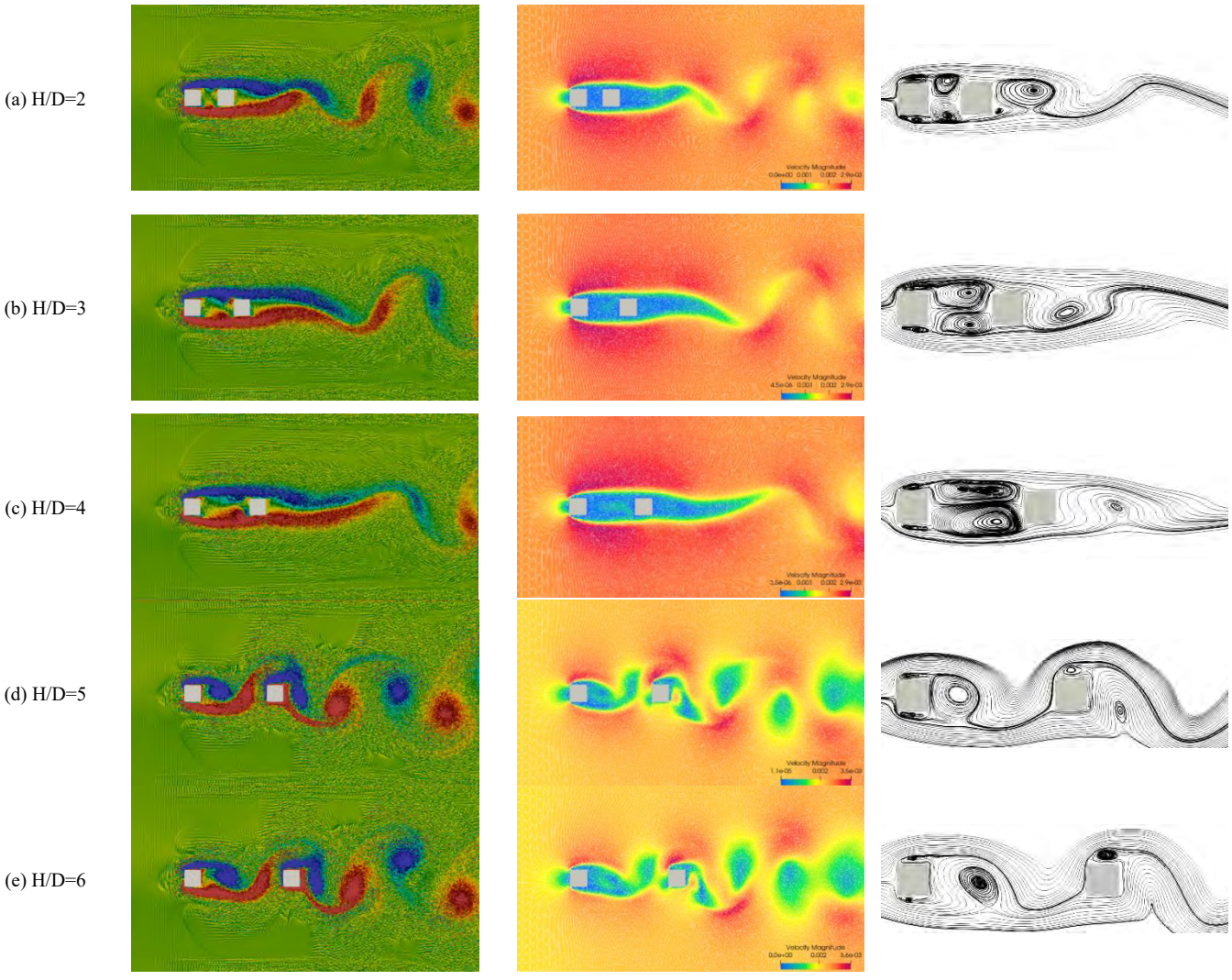


Fig. 5 Vorticity and velocity of different H/D at Ut/D=250

Figure 5 (i) and (ii) shows the vorticity, velocity distributions at the time of Ut/D=250 under different spacing ratios. It can be seen that there is no vortex shedding behind the upstream square column when H/D=2~4. When H/D reaches 5, periodic vortex begins shedding behind the upstream square column. With different spacing ratios, obvious Karman vortex street phenomenon can be seen behind the downstream square column. And compared with the simulation of H/D=5 or 6, when H/D=2~4, the wake of the downstream square column is much longer. It can also be clearly seen that under the conditions of H/D=5 and 6, the vortex

shed by the upstream square column moves to the downstream square column and broken, and a new vortex is generated at the downstream square column and falls off.

Figure 5 (iii) shows the streamline diagram of the fully developed flow at different spacing ratios. When the spacing ratios are 2, 3, and 4, small vortices appear on both sides of the upstream column, and flow separation occurs at the two vertices of the front edge of the square column. Due to the interference of the downstream square column, a pair of stable forms behind the upstream square column and will not fall off. Behind the two square columns, there is a periodic shedding of the wake vortices. When the spacing ratios H/D=5 and 6, unlike when the spacing ratios are 2, 3, and 4, the wake vortex behind the upstream square column begins to fall off. At this time, the influence of the downstream column on the upstream column is relatively small. The wake vortices fall off periodically like when the spacing ratio is small.

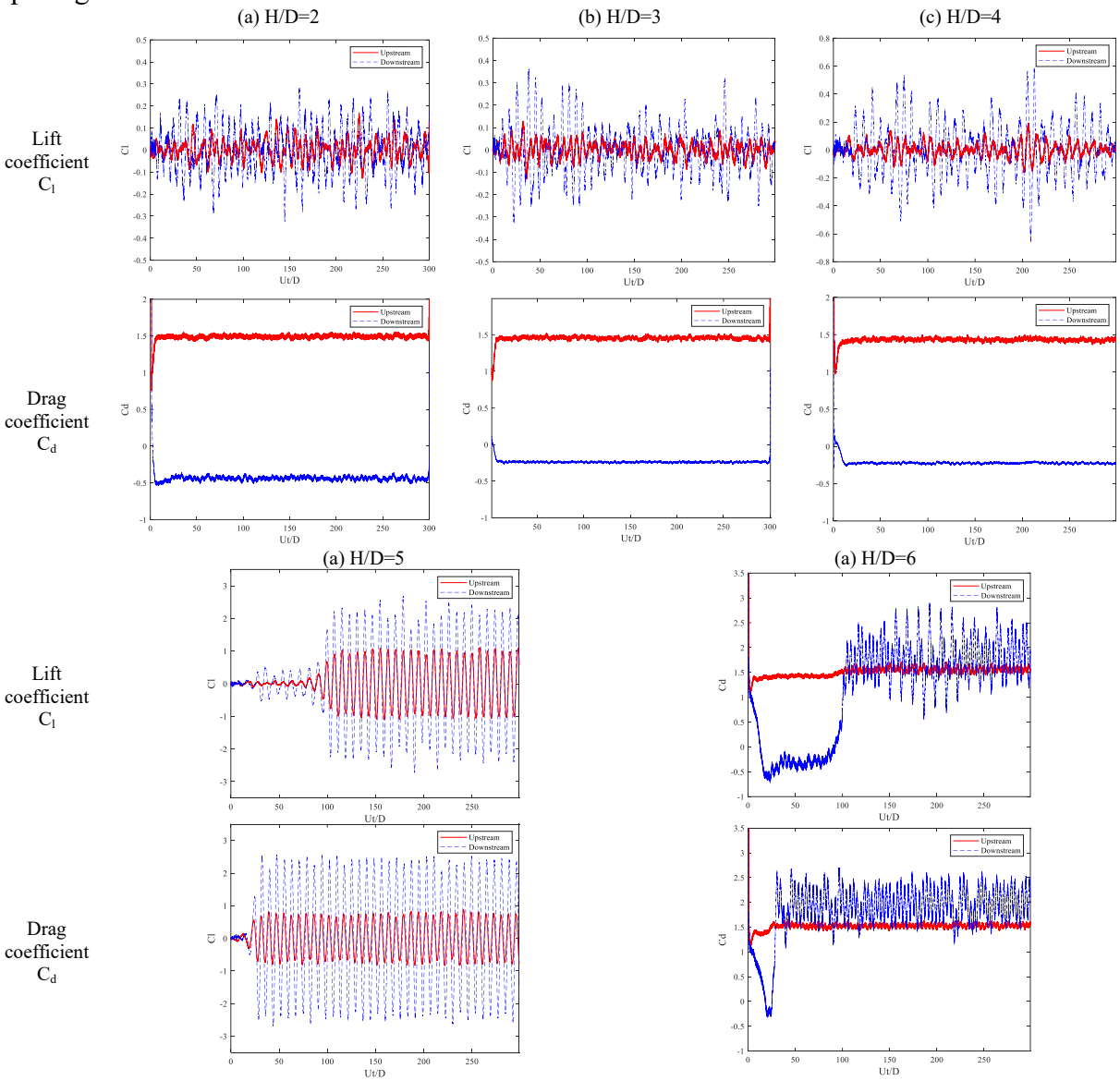


Fig. 6 Time-history curves of lift and drag coefficients with different spacing ratio

Next, the differences in force between the upstream and downstream square columns are analyzed through the lift coefficient and drag coefficient. Figure 6 shows the time-history curves of the lift coefficient and drag coefficient of the upstream and downstream columns at five different spacing ratios of H/D=2~6. It can be seen that the lift coefficient Cl oscillates

obviously at different spacing ratios. When H/D=2~4, the lift coefficient exhibits irregular vibration due to the mutual interference between the columns. When H/D=5 and 6, the oscillation of Cl is stable and the amplitude is relatively large. Regardless of the value of the spacing ratio, the oscillation amplitude of the lift coefficient of the downstream square column is larger than that of the upstream square column. For the drag coefficient Cd, with the increase of the spacing ratio, the drag coefficient of the upstream square column has no obvious change, and its value is around 1.5. For the downstream square column, when H/D=2~4, the drag coefficient is less than zero, and it is around a certain value like the upstream square column. When the spacing ratio reaches 5, the drag coefficient curve of the downstream square column has changed significantly, indicating that the flow state around the downstream column has changed. With the development of the flow, the drag coefficient of the downstream square column increases suddenly to above zero, and there is a large amplitude oscillation. It shows that the downstream square column is affected by the wake of the upstream column, and the change of the flow state makes the drag coefficient of the downstream square column make a big change, from less than zero to greater than zero.

Conclusions

In this paper, a simulation of the flow around two square columns in a row of the fluid coming direction is carried out by the MLParticle-SJTU solver based on the MPS method. Firstly, the accuracy of the solver is verified by simulating the Poiseuille flow. Then, the numerical simulation is carried out on the flow pass the two column in a row with Reynolds number of 200. It is found that vortex shedding does not occur behind the upstream square column when the spacing ratio H/D$\leqslant$4, and vortex shedding occurs behind the upstream square column when the spacing ratio H/D$\geqslant$5. The change of the flow state makes the drag coefficient curve of the downstream square column change greatly, which changes from a negative value to a positive value. Under the condition of H/D$\geqslant$5, the upstream square column will produce periodic vortex shedding. The vortex acts on the downstream square column, so that the lifting resistance coefficient oscillation amplitude of the downstream square column increases greatly.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (52131102), to which the authors are most grateful.

References

[1] Sakamoto, H., Haniu, H. and Obata, Y. (1987). Fluctuating forces acting on two square prisms in a tandem arrangement, *Journal of Wind Engineering and Industrial Aerodynamics*, **26**(1): 85-103

[2] Kim, M. K., Kim, D. K., Yoon, S. H., et al. (2008) Measurements of the flow fields around two square cylinders in a tandem arrangement, *Journal of Mechanical Science and Technology*, **22**(2): 397-407.

[3] Sohankar, A., Etminan, A. (2009) Forced-convection heat transfer from tandem square cylinders in cross flow at low Reynolds numbers, *International Journal for Numerical Methods in Fluids*, **60**(7):733 – 751.

[4] Sohankar, A. (2012) A numerical investigation of the flow over a pair of identical square cylinders in a tandem arrangement, *International Journal for Numerical Methods in Fluids*, **70**: 1244–1257.

[5] Chatterjee, D., Mondal, B. (2012) Forced convection heat transfer from tandem square cylinders for various spacing ratios, *Numerical Heat Transfer, Part A*, **61**(5): 381-400.

[6] Ehsan, I., Mohammad, S., Reza, N. M., et al. (2013) Power-law fluid flow passing two square cylinders in tandem arrangement, *Journal of Fluids Engineering*, **135**(6): 1-8.

[7] Kumar, D., Sourav, K., Sen, S. (2019) Steady separated flow around a pair of identical square cylinders in tandem array at low Reynolds numbers, *Computers and Fluids*, **191**: 1-16.

[8] Zhang Y., Wan, D. (2014) Comparative study of MPS method and level-set method for sloshing flows, *Journal of Hydrodynamics*, **26**(4): 577-585.

[9] Zhang, Y., *Development and Application of 3D Parallel Improved Meshless MPS Method*, PhD Thesis, Shanghai Jiao Tong University, China, 2014.
[10] Zhang, G., Zha, R., Wan, D. (2022) MPS–FEM coupled method for 3D dam-break flows with elastic gate structures, *European Journal of Mechanics / B Fluids*, **94**: 171–189.
[11] Xie, F., Zhao, W., Wan, D. (2021) MPS-DEM coupling method for interaction between fluid and thin elastic structures, *Ocean Engineering*, **236**: 109449.
[12] Chen, X., Wan, D. (2019) GPU accelerated MPS method for large-scale 3-D violent free surface flows, *Ocean Engineering*, **171**: 677–694.
[13] Tang, Z., Zhang, Y., Wan, D. (2016) Multi-resolution MPS method for free surface flows, *International Journal of Computational Methods*, **13**(4): 1641018.
[14] Huang, C., Zhang, G., Zhao, W., Wan, D. (2022) Hydroelastic responses of an elastic cylinder impacting on the free surface by MPS-FEM coupled method, *Acta Mechanica Sinica*, **38**(11): 322057.
[15] Tanaka, M., Masunaga, T. (2010) Stabilization and smoothing of pressure in MPS method by quasi-compressibility, *Journal of Computational Physics*, **229**(11): 4279- 4290.
[16] Lee, B., Park, J., Kim, M., Hwang, S. (2011) Step-by-step improvement of MPS method in simulating violent free-surface motions and impact-loads, *Computer Methods in Applied Mechanics and Engineering*, **200**(9-12): 1113-1125.
[17] Brackbill, J. U., Kothe, D. B., Zemach, C. (1992) A continuum method for modeling surface tension, *Journal of Computational Physics*, **100**(2): 335-354.

Numerical Simulation of Hydrodynamic and Thermal Wakes of an Underwater Vehicle

Gang Gao[1*], Yangjun Wang[2], Liushuai Cao[1], Decheng Wan[1†]

[1] Computational Marine Hydrodynamic Lab (CMHL), School of Naval Architecture, Ocean and Civil Engineering, Shanghai Jiao Tong University, Shanghai, China
[2] College of Advanced Interdisciplinary Studies, National University of Defense Technology, Nanjing, China

*Presenting author: GC911741@126.com
†Corresponding author: dcwan@sjtu.edu.cn

Abstract

During the movement of underwater vehicle, hydrodynamic wake are formed along with long-distance thermal wake caused by the release of thermal wastewater. By capturing and imaging the thermal wake generated behind underwater vehicle in real-time, it is possible to determine the size, speed, and location of the vehicle. Investigating the hydrodynamic wake and thermal wake of underwater vehicle requires the integration of various physical fields, including fluid mechanics and temperature. Current research is centered on analyzing the impact of factors like the underwater vehicle's speed, temperature and flow rate of thermal wastewater on the wake. However, there is a lack of comprehensive studies in this area, as well as limited attention given to the effects of attachments and propellers on thermal wake. This study conducted numerical simulations using the RANS method to analyze the hydrodynamic and thermal wake of an underwater vehicle with a fully attached body. The research considered various speeds, propeller speeds, temperatures and flow rates of thermal wastewater. The results revealed that the propeller significantly influences the core region of the thermal wake. The temperature decays rapidly when closer to the vehicle and slower at greater distances. Moreover, higher speeds and propeller rotation speeds lead to a larger affected area by temperature. Additionally, increasing the temperature and flow rate of thermal wastewater strengthens the thermodynamic signal in the wake behind the underwater vehicle.

Keywords: hydrodynamic wake; thermal wake; numerical simulation; underwater vehicle; thermal wastewater

Introduction

The continuous operation of the engine is used to propel underwater vehicles, while cooler water within the vehicle is employed to cool the heating engine. This water, after absorbing heat from the engine, experiences a temperature increase. The underwater vehicle then discharges this heated waste water into the surrounding environment. The thermal waste water exchanges heat with the ambient water, creating a thermal signal behind the underwater vehicle and at the free surface. Cao et al. (2024,2023) pointed out that this thermal signal can be detected by infrared equipment, which allows for the estimation of the underwater vehicle's size, position, and speed. Utilizing thermal signals for such detection has become a crucial method for identifying and tracking underwater vehicles [1-2].

There has been significant research on the hydrodynamic and thermal wakes generated by underwater vehicles. In an experimental study, Luo et al. (2024) established a testing platform

in an outdoor pool to investigate the thermal wake of underwater vehicles. Using infrared imaging, obtained parameters and images of the thermal wake at the water surface. Results revealed that vortices of different scales within the wake interact with the water surface's thermal boundary layer, leading to modulation, amplification, and dispersion effects that result in variations in temperature characteristics [2]. Similarly, Wang et al. (2023) conducted experiments to analyze the effects of parameters such as Reynolds number, submersion depth, propeller speed, thermal jet, and temperature difference on the temperature distribution at the free surface. Findings indicated that the infrared signals on the water surface are not direct thermal signals generated by the buoyancy of the thermal wake. Also observed that the area of temperature signal on the water surface increases with greater submersion depth [3]. Li et al. (2023) developed an experimental platform for investigating thermal wakes using a scaled-down Suboff model. Through tests of straight-line motion and thermal discharge, characterized the temperature distribution at the free surface [4].

Experimental studies on underwater vehicles predominantly use scaled models, as conducting full-scale experiments under laboratory conditions poses significant challenges. In terms of numerical simulations, Luo et al. (2023) developed a thermal wake computation model based on the Detached Eddy Simulation (DES) method. And investigated the interaction between the hydrodynamic wake and the temperature boundary layer, focusing on thermal diffusion and heat transfer at the interface. Findings indicated that the thermal wake on the free surface exhibits a distinct elongated vortex structure [5]. Gu et al. (2018), utilizing the Volume of Fluid (VOF) method and overset grid technique, simulated the thermal wake of an underwater vehicle under conditions of density and temperature stratification. And discovered that both thermal and hydrodynamic wakes could manifest on the free surface. In deeper regions, colder water was observed to ascend to the surface due to the disturbance caused by the underwater vehicle, forming a cold wake [6].

Zhang et al. (2019) employed the finite volume method to analyze the impact of submersion depth, speed, and propeller rotation on the thermal wake using the LSII submarine model. Results demonstrated that disturbances in cold seawater at greater depths lead to the formation of a cold wake. The interaction of cold and thermal wakes creates a unique downstream surface pattern characterized by alternating cold and warm wakes [7]. Huang et al. (2024) established a coupled model of flow and electromagnetic fields to study the evolution of the far-field wake magnetic field of a full-scale submarine [8]. Chen et al. (2023) developed an interdisciplinary model incorporating continuity, momentum, heat, mass, and Maxwell's equations to conduct numerical simulations of the velocity, vorticity, and electromagnetic field distribution in the far field of a full-scale submarine [9].Luo et al. (2022) further investigated the effects of thermal jets on the water body, discovering that thermal conduction, convection, and entrainment contribute to the presence of thermal signals, cold-thermal signals, and cold signals across different water layers. The boundaries and distribution range of these signals in various layers are determined by the parameters of the thermal jet and the submersion depth [10].

It is evident that both experimental and numerical simulation studies have predominantly focused on observing thermal signals near the free surface. There is relatively less attention given to the thermal signals behind underwater vehicles at greater depths. Furthermore, most research has been conducted using scaled models rather than full-scale underwater vehicles. This study addresses the thermal wake of a full-scale underwater vehicle with complete appendages. Based on the Reynolds-Averaged Navier-Stokes (RANS) method, conducted numerical simulations under various conditions, including different vehicle speeds, propeller

rotations, and thermal wastewater temperatures and flow rates. The structure of this paper is as follows: begin with an introduction to the numerical simulation methods employed in this study. And then present the computational results for typical scenarios, followed by a discussion of the corresponding conclusions.

Numerical method

Governing equations

In the solution process of this paper, the fluid motion satisfies the mass conservation and momentum conservation equations.

$$\nabla \cdot \mathbf{U} = 0 \tag{1}$$

$$\frac{\partial \rho \mathbf{U}}{\partial t} + \nabla \cdot (\rho \mathbf{U}\mathbf{U}) = -\nabla p + \rho g + \nabla \cdot (\mu \nabla \mathbf{U}) \tag{2}$$

where U is the velocity field, ρ is the density of fluid, p is the pressure field, g is the gravity acceleration, μ is the dynamic viscosity coefficient. Turbulence model using *SST* $k-\omega$ model,

Geometric Model

The SUBOFF (Submarine Fully Appended) model used in this study is a standardized submarine model developed by the U.S. Defense Advanced Research Projects Agency (DARPA), providing reliable comparative data for numerical simulations. The model consists of a streamlined hull, an outer shell, and four symmetrical tail fins. It has a length of 4.356 meters and a diameter of 0.508 meters. For this study, the model is scaled up by a factor of 23, and the propeller is modeled using the E1619 propeller model. The geometric representation of the model is shown in Figure 1, table1 lists the key parameters of the SUBOFF submarine and the propeller.

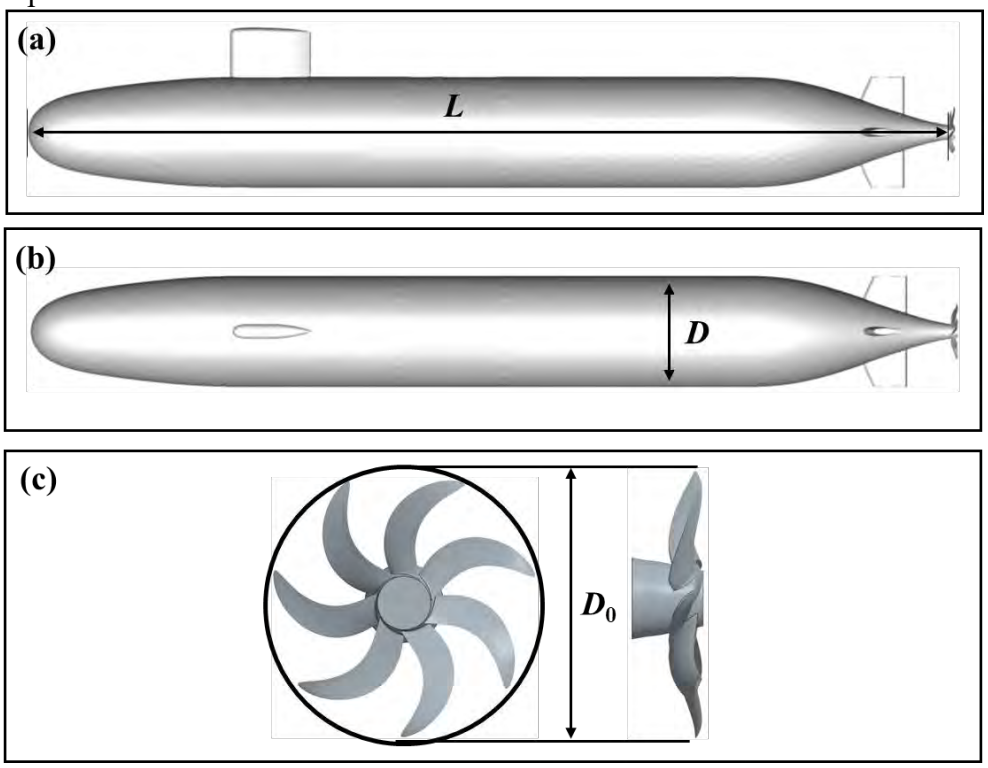


Figure 1. The SUBOFF fully appended submarine model

Table 1. Main parameters of the submarine-propeller system at full scale

Parameter	Symbol	Unit	Value
Hull length	L	m	100.188
Hull diameter	D	m	11.684
Propeller diameter	D_0	m	9.564
Number of blades	N	-	7

Computational domain setup

The computational domain setup is illustrated in Figure 2. The inlet of the computational domain is positioned 1L upstream from the nose of the underwater vehicle. The lateral boundaries are set 1.5L from the sides of the vehicle. The outlet is placed 11L downstream from the stern of the vehicle. The top boundary is located 1L above the vehicle, while the bottom boundary is 0.5L below the vehicle.

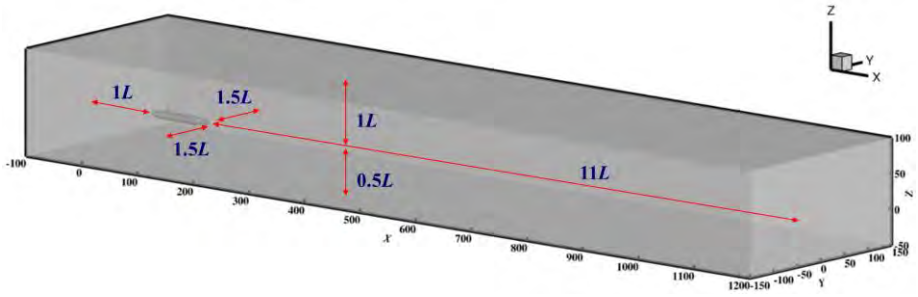


Figure 2. Computational domain setup

Mesh generation

As shown in Figure 3, local grid refinement is applied in regions with high surface curvature on the underwater vehicle and near the thermal discharge port. Additionally, a conical refinement is implemented in the wake region to account for the upward movement of the wake during its development.

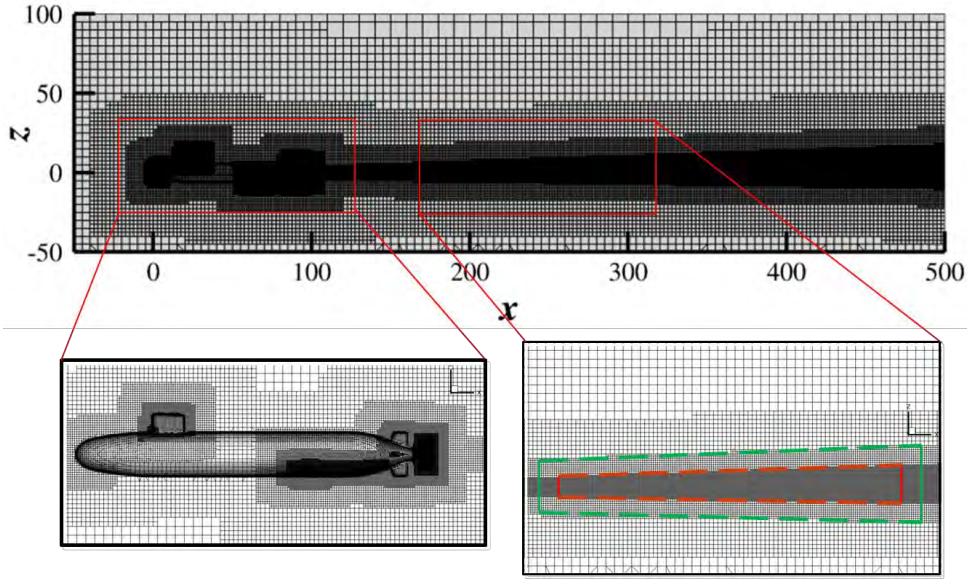


Figure 3. Computational mesh

Boundary condition

In terms of boundary condition settings, the inlet of the computational domain is defined as a velocity inlet boundary, and the outlet is set as a pressure outlet boundary. The lateral and vertical boundaries of the computational domain are all specified as velocity inlet boundaries.

Computational conditions setup

As shown in Table 2, the operating conditions for the calculations in this study are presented, including the velocity of the underwater vehicle, the rotational speed of the propeller, the temperature and flow rate of the thermal discharge water, and the temperature of the ambient seawater.

Table 2. Computational conditions

Case	The speed of underwater vehicles(*kn*)	The speed of the propeller(*rpm*)	Temperature difference(*K*)	Floe rate (*t/h*)	The temperature of the background(*K*)
1	6	42	14	450	277.51
2	12	84	17	900	277.51
3	18	126	20	1350	277.51
4	30	210	23	2250	277.51

Results and Discussions

The temperature distribution in the wake

Firstly, the changes in temperature distribution in the wake at different cruising speeds were compared, as illustrated in Figure 4. The figure presents the variation curves of the maximum temperature at various positions behind the underwater vehicle under different cruising speeds. Due to the initial condition settings, it can be observed that as the cruising speed increases, the maximum temperature also rises. Additionally, as the distance from the vehicle increases, the maximum temperature decreases for all cruising speeds. The rate of temperature decay becomes more pronounced with higher cruising speeds at larger distances.

Figures 5 and 6 respectively show the temperature distribution on the z=0 and y=0 planes. It is evident that the temperature decreases as the distance from the underwater vehicle increases. In terms of distribution range, the vertical temperature distribution spans a greater extent compared to the lateral distribution. Moreover, as the distance from the underwater vehicle increases, the temperature distribution in the wake begins to exhibit oscillations. For example, within the range of ($x/L > 6$), a spindle-shaped distribution pattern starts to emerge.

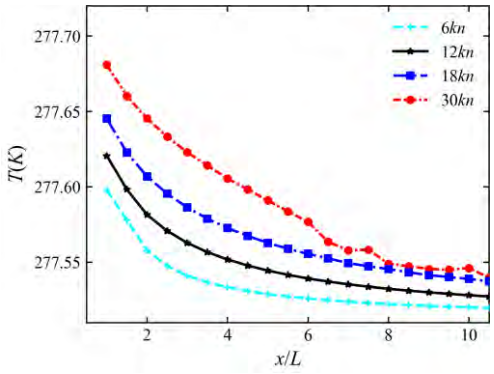


Figure 4. Temperature maximum value variation curve of the wake

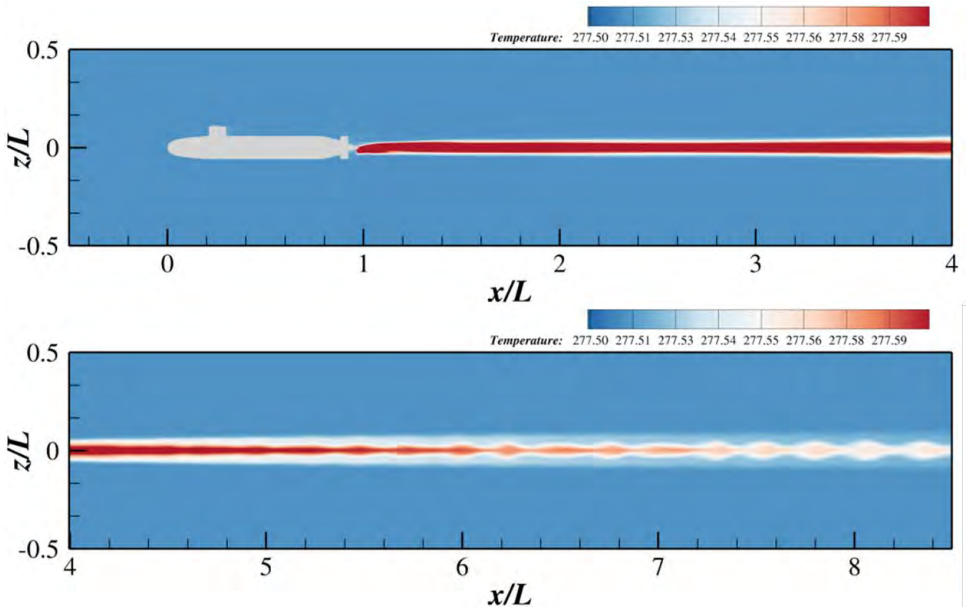


Figure 5. Temperature distribution of the wake (z=0, 30kn)

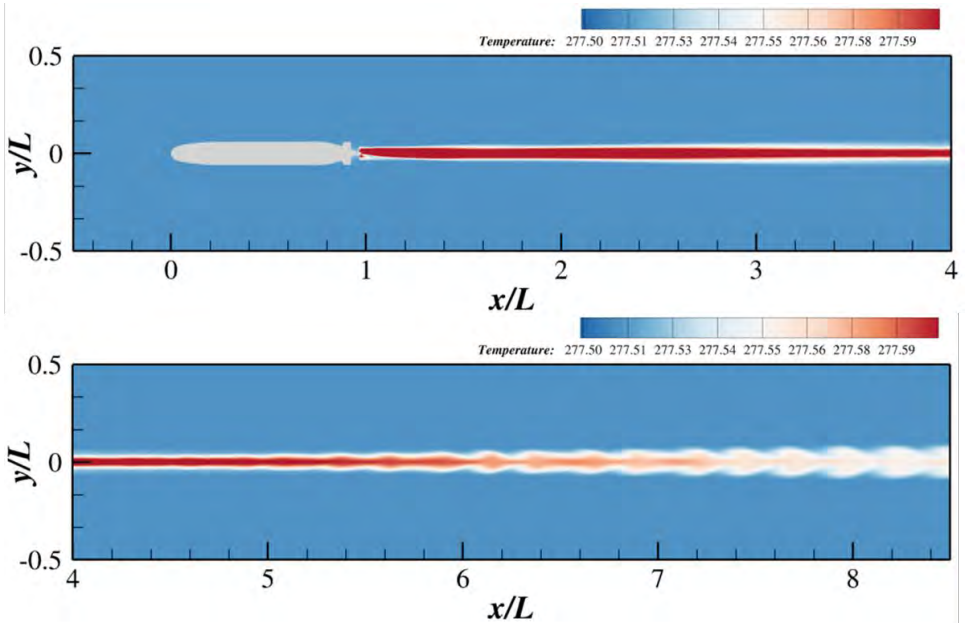


Figure 6. Temperature distribution of the wake (y=0, 30kn)

Figure 7 presents the three-dimensional structure of temperature iso-surface, specifically set at 277.55 K. Upon closer examination of the enlarged local view, it becomes evident that these iso-surface primarily depict the thermal wake generated by the thermal discharge from the underwater vehicle's outlet. As the distance from the underwater vehicle increases, the overall shape of the iso-surface exhibits a spiraling pattern. Observing the temperature distribution plots of different sections behind the underwater vehicle in Figure 8, it is noticeable that as the distance increases, the temperature values in the wake gradually decrease. Simultaneously, the area affected by the temperature of the wake expands. Between x/L = 3 and x/L = 7, the temperature distribution in the wake forms two trailing vortex tips, closely resembling the topology of the propeller. This indicates significant influence of the propeller on the temperature distribution of the wake. However, from x/L = 8 to x/L = 9, the variation in the

temperature distribution of the wake becomes minimal, suggesting a reduced disturbance effect of the propeller in this range.

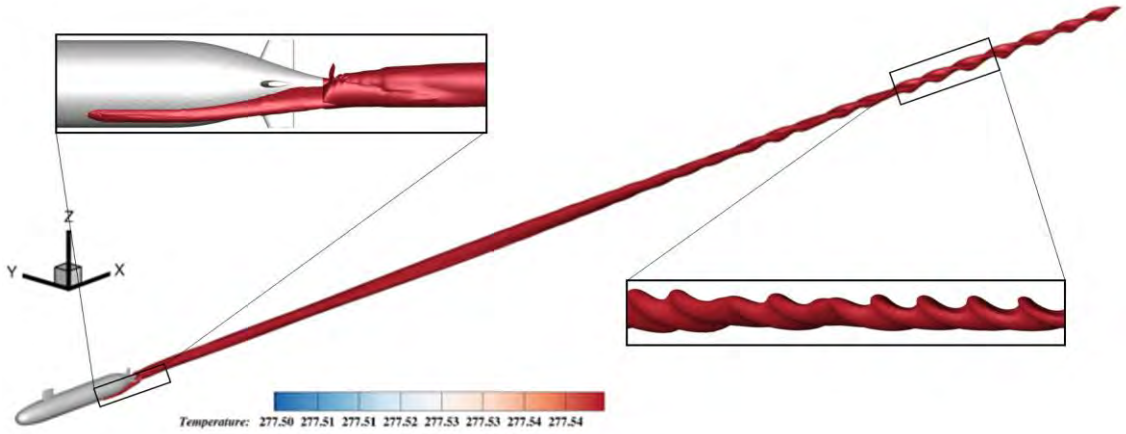


Figure 7. Temperature iso-surface of the wake (*T*=277.55*K*, 30*kn*)

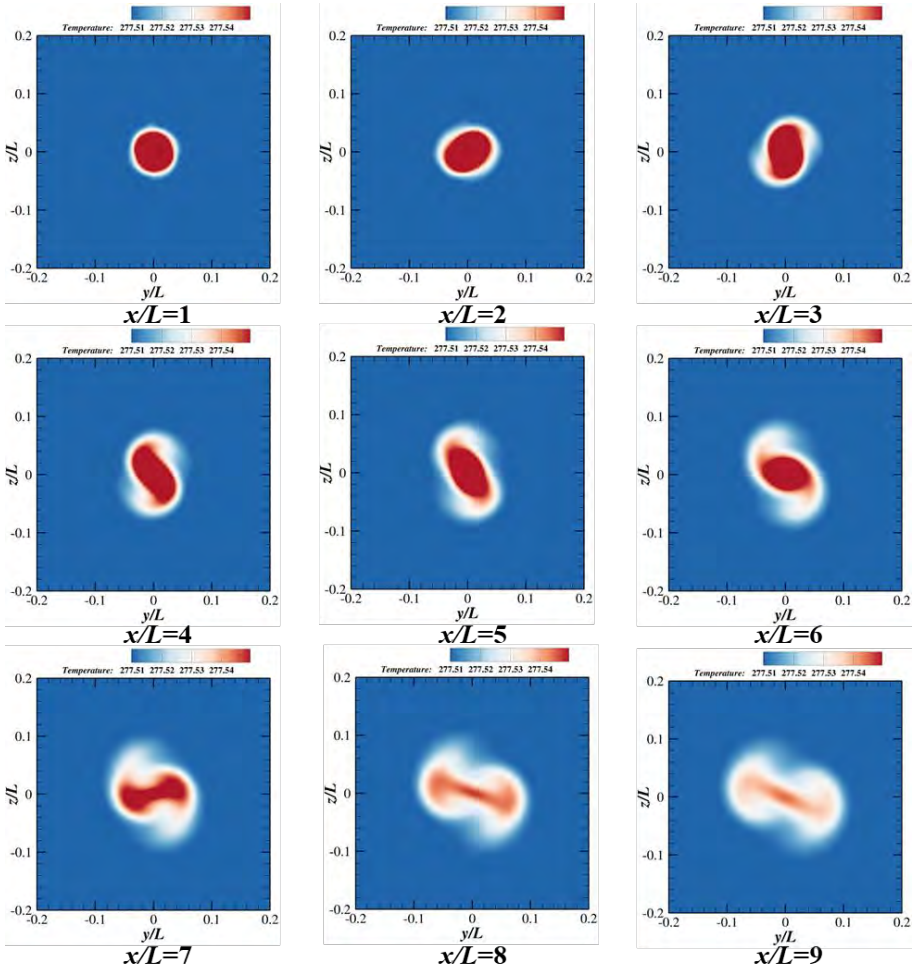


Figure 8. Distribution of temperature (30*kn*)

The velocity distribution in the wake

Subsequently, the variations in velocity distribution in the wake at different cruising speeds were compared. Figure 9(a) presents the curves depicting the changes in maximum velocity at various positions behind the underwater vehicle under different cruising speeds. It is evident

that as the cruising speed increases, the maximum velocity also increases. Additionally, as the distance increases, the maximum velocity decreases gradually for all cruising speeds. Figure 9(b) illustrates the normalized velocity variation curves, showing that with increasing cruising speed, the decay rate of maximum velocity becomes more pronounced at larger distances.

Figures 10 and 11 respectively illustrate the velocity distributions on the z=0 and y=0 planes. It can be observed that the velocity values decrease as the distance from the underwater vehicle increases. Similarly, in terms of distribution range, the vertical distribution of velocity spans a significantly larger extent compared to the lateral distribution. Furthermore, as the distance from the underwater vehicle increases, the velocity distribution in the wake begins to exhibit oscillations. For instance, within the range of $x/L > 6$, a spindle-shaped distribution pattern also emerges.

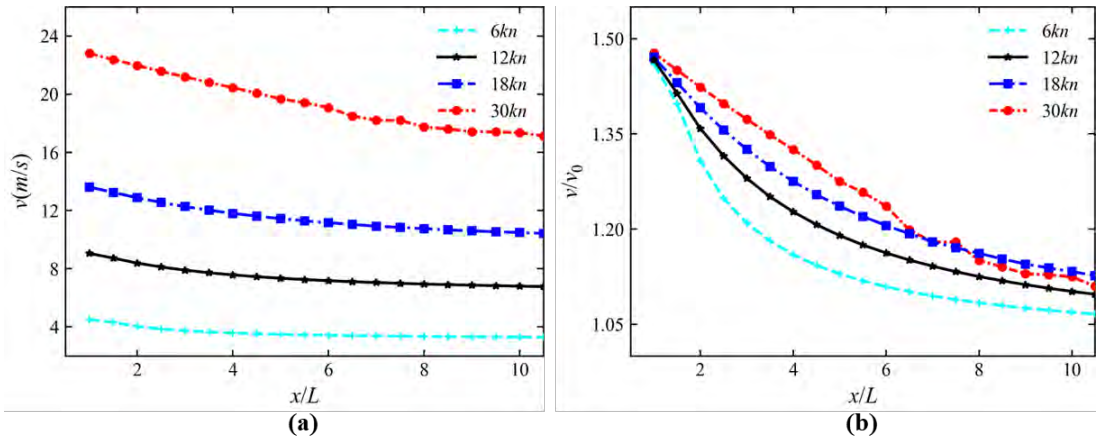


Figure 9. Velocity maximum value variation curve of the wake

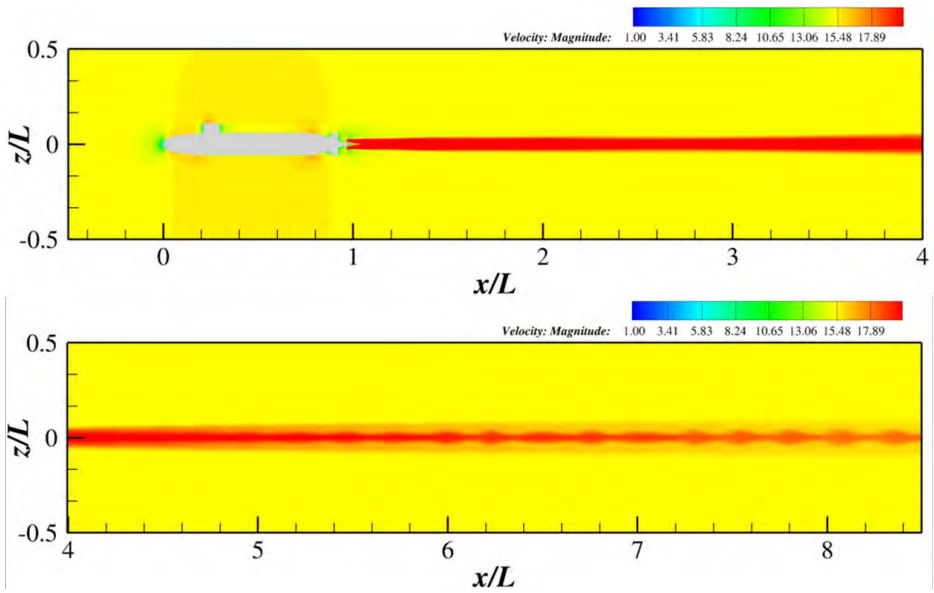


Figure 10. Velocity distribution of the wake (z=0, 30kn)

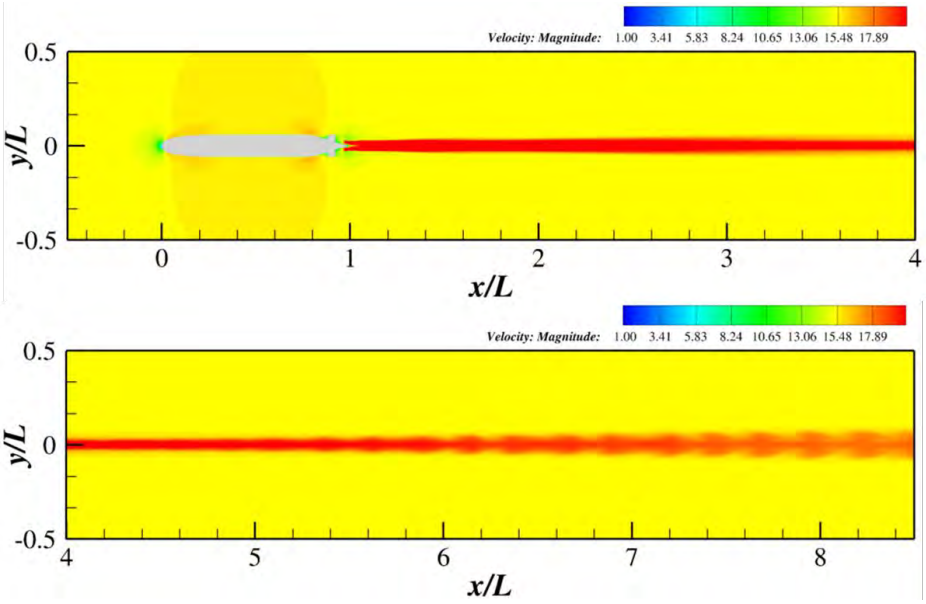


Figure 11. Velocity distribution of the wake (*y*=0, 30*kn*)

Figure 12 depicts the three-dimensional structure of velocity iso-surface, set at a value of 18 m/s. Upon closer examination of the enlarged local view, it becomes evident that the iso-surface expand gradually when closer to the underwater vehicle, while assuming a spiraling pattern as the distance from the underwater vehicle increases. Figure 13 illustrates the velocity distribution plots of different sections behind the underwater vehicle. It can be observed that as the distance increases, the velocity values in the wake gradually decrease. Simultaneously, the area affected by the wake's velocity expands. Between x/L = 3 and x/L = 7, the velocity distribution in the wake exhibits two trailing vortex tips, resembling the topology of the propeller. This indicates a significant influence of the propeller on the velocity distribution of the wake. However, from x/L = 8 to x/L = 9, the variation in the velocity distribution of the wake becomes minimal, suggesting a reduced disturbance effect of the propeller in this range.

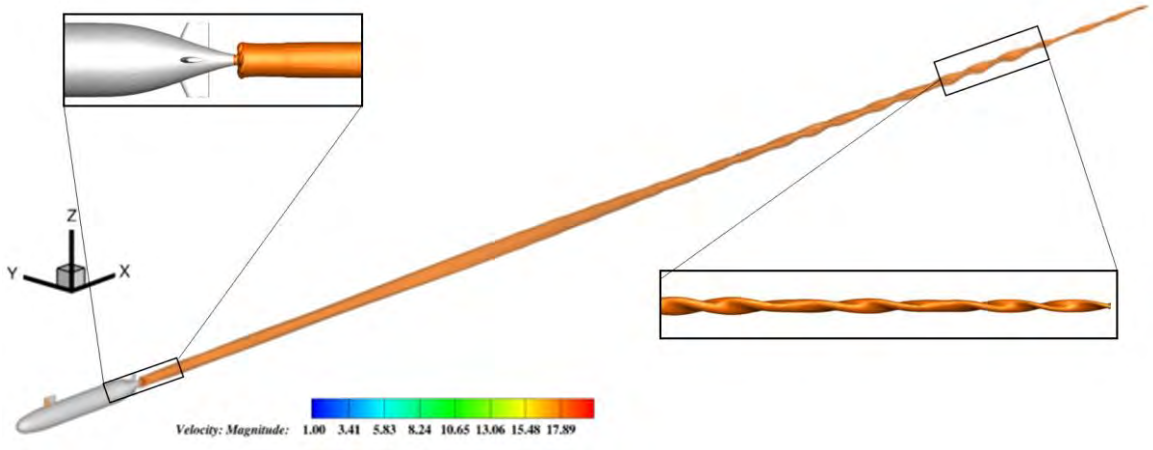


Figure 12. Velocity iso-surface of the wake (*v*=18m/s, 30*kn*)

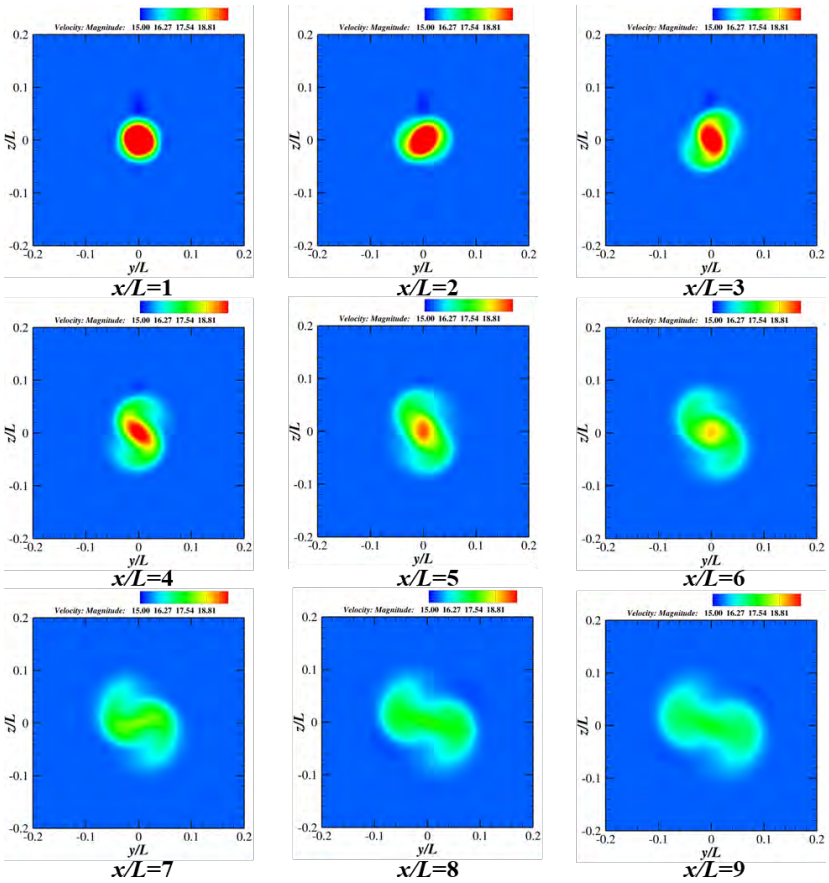


Figure 13. Distribution of velocity (30*kn*)

The vorticity distribution in the wake

Subsequently, the variations in vorticity distribution in the wake at different cruising speeds were compared. Figure 14(a) presents the curves depicting the changes in maximum vorticity at various positions behind the underwater vehicle under different cruising speeds and propeller rotational speeds. It can be observed that as the cruising speed and propeller rotational speed increase, the maximum vorticity also increases. Additionally, as the distance increases, the maximum vorticity decreases gradually for all cruising speeds. Figure 14(b) illustrates the normalized vorticity variation curves, showing that with decreasing cruising speed, the decay rate of maximum vorticity becomes more pronounced at larger distances.

Figures 15 and 16 respectively illustrate the velocity distributions on the z=0 and y=0 planes. It can be observed that the velocity values decrease as the distance from the underwater vehicle increases. For instance, within the range $x/L > 4$, the maximum vorticity in the wake predominantly surrounds the underwater vehicle rather than being concentrated along its axis.

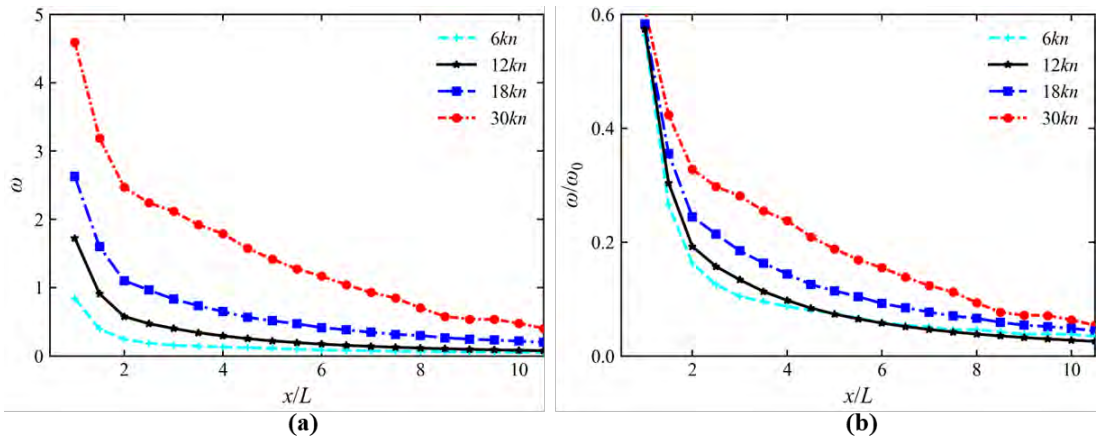


Figure 14. Vorticity maximum value variation curve of the wake

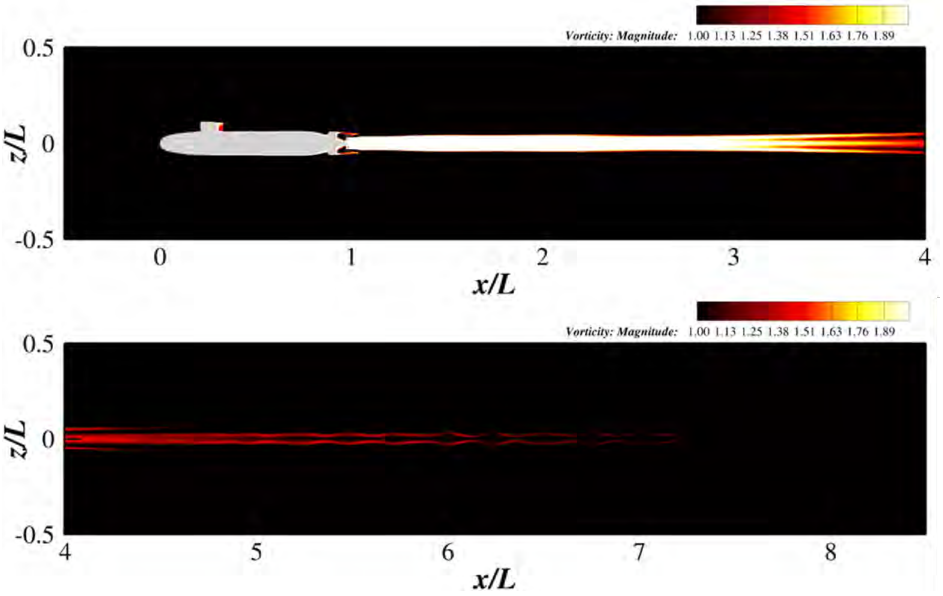


Figure 15. Vorticity distribution of the wake (*z*=0, 30*kn*)

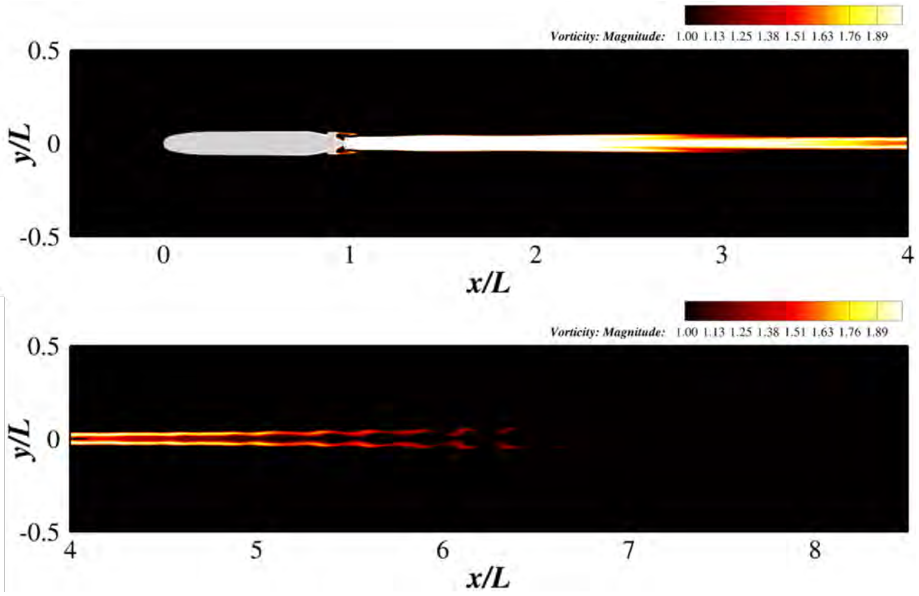


Figure 16. Vorticity distribution of the wake (*y*=0, 30*kn*)

Figure 17 presents the three-dimensional structure of vorticity isosurfaces, set at a value of 0.5 s^{-1}. Upon closer examination of the enlarged local view, it is evident that significant attachment vortices exist on the surface of the underwater vehicle. As the distance from the underwater vehicle increases, the overall shape of the isosurfaces also exhibits a spiraling pattern. Observing the vorticity distribution plots of different sections behind the underwater vehicle in Figure 18, it can be observed that as the distance increases, the vorticity in the wake gradually decreases.

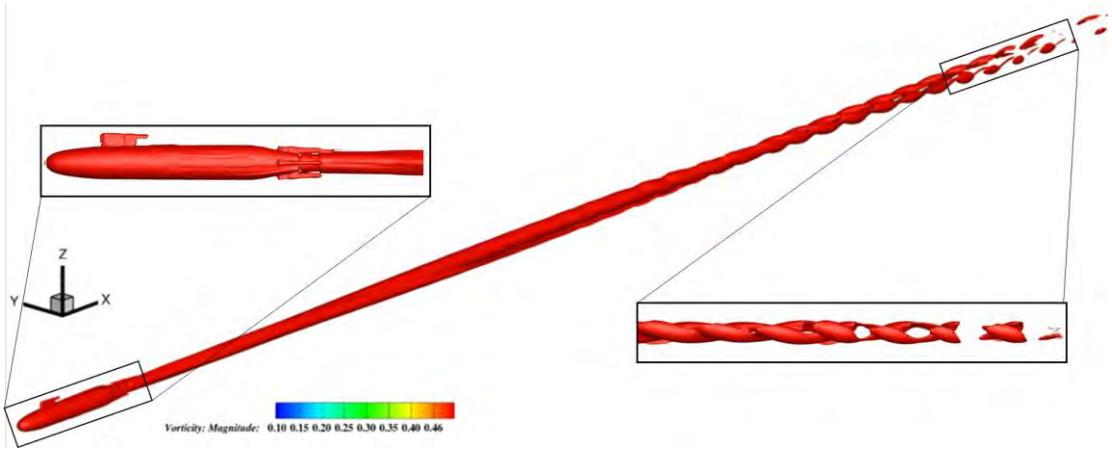


Figure 17. Vorticity iso-surface of the wake (*w* =0.5/*s*, 30*kn*)

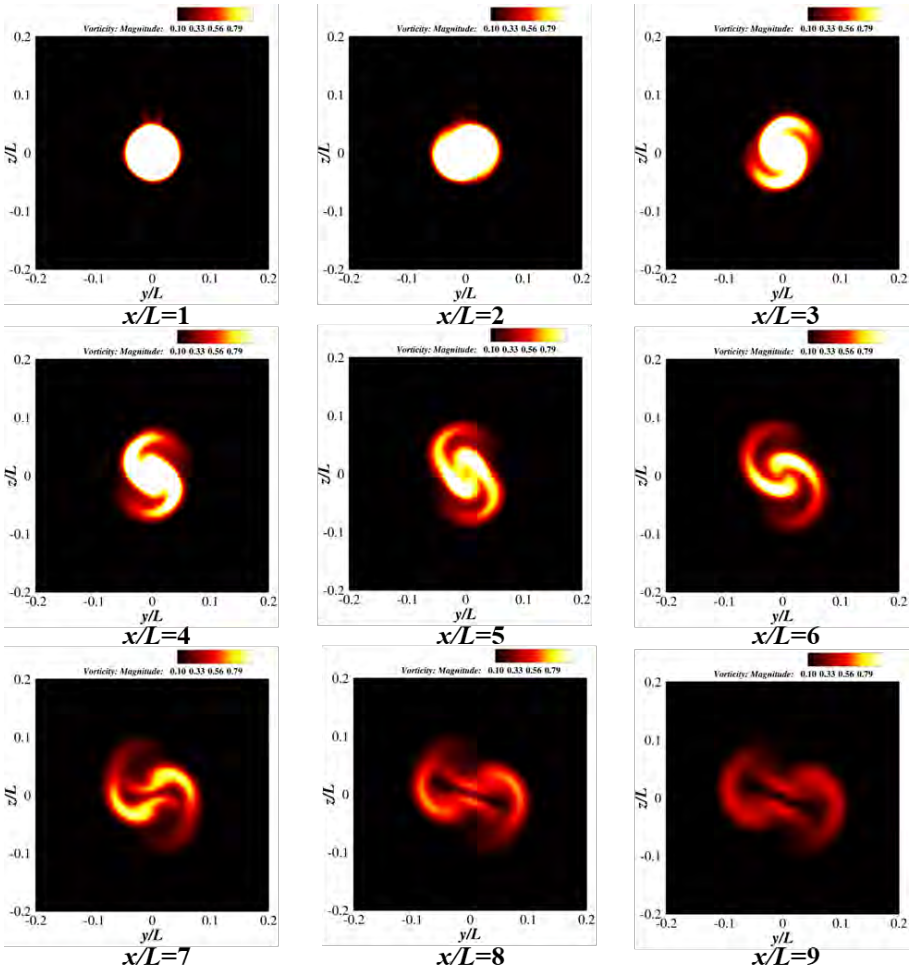


Figure 18. Distribution of vorticity (30*kn*)

Conclusions

Based on the RANS (Reynolds-Averaged Navier-Stokes) method and using the *SST k-ω* turbulence model, this study conducted numerical simulations of hydrodynamic wakes and thermal wakes at full scale and various cruising speeds, using the SUBOFF model as the subject. The conclusions are as follows:

(a) For the temperature, velocity, and vorticity values in the wake, increase with increasing cruising speed and gradually decrease with increasing distance from the underwater vehicle.

(b) For the temperature and velocity distributions in the wake, the vertical distribution range is greater than the lateral distribution range. The iso-surface of temperature, velocity, and vorticity in the wake exhibit a spiral pattern at greater distances from the underwater vehicle.

(c) For the temperature, velocity, and vorticity distributions at different cross-sections behind the underwater vehicle, the values of these physical quantities decrease as the distance increases, while their influence on the surrounding fluid expands. There are two trailing vortices present, which have a topological relationship with the shape of the propeller.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (52131102), to which the authors are most grateful.

References

[1] Cao, L., Pan, Y., Gao, G., Li, L., & Wan, D. (2024). Review on the Hydro-and Thermo-Dynamic Wakes of Underwater Vehicles in Linearly Stratified Fluid. Journal of Marine Science and Engineering, 12(3), 490.

[2] Cao, L., Gao, G., Guo, E., & Wan, D. (2023). Hydrodynamic performances and wakes induced by a generic submarine operating near the free surface in continuously stratified fluid, Journal of Hydrodynamics, 35(3): 396-406.

[3] Luo, F., Shuai, C., Du, Y., Gao, C., & Wang, B. (2024). Experimental on the thermal characteristics of surface wake generated by submerged vehicle. Ocean Engineering, 297, 116957.

[4] Wang, C. A., Xu, D., & Gao, J. P. (2023). Surface temperature characteristics of underwater thermal jet based on thermal skin. Applied Ocean Research, 130, 103411.

[5] Li, G., Du, Y., & Yang, L. (2023). Simulation study on thermal wake characteristics of underwater vehicle under rotary motion. Applied Sciences, 13(3), 1531.

[6] Luo, F., Shuai, C., Du, Y., & Ma, J. (2023). Thermal characteristics of vehicle wake induced by the interaction between hydrodynamic wake and cold skin. Ocean Engineering, 267, 113272.

[7] Gu, W., Zhang, J. L., Meng, Q. J., Peng, L., Shen, Z. B., & Deng, H. H. (2018, June). Study of Underwater Vehicle's Wake in Seawater of Linearly Gradient Temperature and Density. In ISOPE International Ocean and Polar Engineering Conference (pp. ISOPE-I). ISOPE.

[8] Zhang, X., Guo, L., Hu, R., & Liu, C. (2019, March). Cold-thermal wake characteristics of submarine in temperature-density stratified seawater. In Fifth Symposium on Novel Optoelectronic Detection Technology and Application (Vol. 11023, pp. 1293-1298). SPIE.

[9] Huang, B., Liu, Z., Xu, Y., Pan, M., Hu, J., & Zhang, Q. (2024). Far-field characteristics and evolutions of electromagnetic field induced by the wake of underwater vehicles. Applied Ocean Research, 145, 103933.

[10] Chen, Q., Xuan, Y., Lin, Q., Han, Y., & Wei, K. (2023). Evolutions of hydrodynamic and electromagnetic wakes induced by underwater vehicles. Applied Ocean Research, 140, 103750.

[11] Luo, F., Shuai, C., Du, Y., & Ma, J. (2022). Numerical simulation of temperature variation in water layer induced by thermal jet in weak stratified environment. AIP Advances, 12(6).

An Effective Out-of-core Automated Multi-Level Substructuring Method for Dynamic Analysis of Large-scale Structures

***Guidong Wang[1], †Xiangyang Cui[1,2], and Xin Hu[1,2]**

[1]State Key Laboratory of Advanced Design and Manufacturing for Vehicle Body, Hunan University, Changsha, 410082, China.
[2]Hunan Maixi Software Co., Changsha, 410082, China

*Presenting author: hnuwgd@hnu.edu.cn
†Corresponding author: cuixy@hnu.edu.cn

Abstract

A novel out-of-core automated multi-level substructuring (AMLS) method is proposed to address the issue of excessive memory consumption in large-scale structural dynamic analysis. In present method, the nodes in the substructure tree are classified into three categories, and different assembly and computation methods are designed separately. On this basis, various types of intermediate data such as constraint matrices and update matrices of substructures are stored on disk in the solution steps of AMLS. Additionally, a set of data structures has been designed to allow each participating thread to reuse memory space across different substructure computations. This mitigates potential issues such as memory fragmentation and thread blocking during parallel computation. Through this approach, the out-of-core method not only maintains the computational efficiency of AMLS, but also significantly reduces the peak memory. Several numerical examples are employed to validate the efficiency and scalability of the out-of-core method. The results demonstrate that compared to in-core parallel computing, this approach can reduce memory usage by half with a little performance loss.

Keywords: Automated multi-level substructuring (AMLS) method, Out-of-core

Introduction

The automated multi-level substructuring (AMLS) method was initially proposed by Professor Bennighof's team at the University of Texas at Austin and has undergone rapid development over the past three decades [1]-[6]. AMLS method is essentially an extension of the component mode synthesis (CMS) method. It utilizes graph partitioning algorithms to reorganize and partition finite element model matrices into multiple substructures. By synthesizing the modal information of these substructures, AMLS method accurately captures the dynamic characteristics of the overall model. Compared to Lanczos and subspace iteration algorithms, AMLS method significantly reduces the computational time required for finite element dynamic analysis. In addition, the algorithm contains numerous computational tasks that can be processed at the same time, providing many opportunities for parallel computation. Currently, AMLS is widely applied in various engineering fields, including structural dynamic analysis, structural optimization design, vibration control, and fluid-structure interaction problems [7]-[10].

However, as the scale of finite element simulation calculations continues to expand, traditional methods often encounter efficiency reductions due to memory fragmentation

during computation. Moreover, there are instances where the memory capacity of a single computer proves insufficient to meet the demands of large-scale model simulations.

To address these challenges, we conducted a comprehensive analysis of the AMLS solution process and its data dependencies, leveraging insights from out-of-core techniques employed in other algorithms [11]-[12]. Building on this groundwork, we propose a novel out-of-core AMLS method. In our approach, nodes within the substructure tree are classified into three types, each with customized assembly and computation methods. This classification optimizes the management of intermediate data, such as constraint mode matrices and update matrices, which are stored on disk to reduces the peak memory. Furthermore, we have implemented efficient data structures that facilitate memory reuse across various substructure computations, addressing issues such as memory fragmentation and thread blocking during parallel execution.

Methodology

In finite element structural dynamic analysis, the AMLS method is primarily employed to find partial eigenpairs of the eigenvalue problem:

$$Kx = \lambda Mx \tag{1}$$

within a specified frequency range $[f_{\min}, f_{\max}]$. Here, $K \in \mathrm{R}^{nxn}$ denotes the stiffness matrix, and $M \in \mathrm{R}^{nxn}$ represents the mass matrix, both of which are real symmetric and positive definite or semi-definite.

AMLS method

The AMLS method for solving Eq. (1) is divided into four stages: automated partitioning, model transformation, solving the reduced eigenvalue problem, and back transformation. During the automated partitioning stage, the entire model is divided into a large number of substructures. Different from traditional component mode synthesis methods, these substructures in AMLS are automatically partitioned using the nested dissection algorithm, which exploits the sparsity of finite element matrices. This process can be performed using the freely available METIS software package [13]. Figure 1 illustrates the substructure tree diagram generated after two rounds of binary partitioning of a model.Each node in the figure represents a substructure.

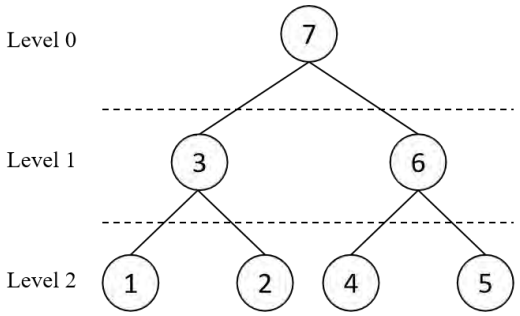


Figure 1. Two-level AMLS substructure tree

Building on this, computations for each node in the substructure tree are performed in a post-order traversal sequence. This process, known as model transformation, is detailed in Algorithm 1. Here, symbols A_i and D_i represent the sets of ancestor nodes and descendant nodes of node i respectively. The symbol $\tilde{A}_i$ is used to denotes the set formed by the union of set A_i and node i.

Algorithm 1. Transformation of node i

1 **for** $j \in \widetilde{A}_i$
2 *Assemble matrices* $\boldsymbol{K}_{ij}$ *and* $\boldsymbol{M}_{ij}$
3 *computer eigenpairs* $\boldsymbol{\Phi}_i$ *and* $\boldsymbol{\Lambda}_i$: $\boldsymbol{K}_{ii}\boldsymbol{\Phi}_i = \boldsymbol{M}_{ii}\boldsymbol{\Phi}_i\boldsymbol{\Lambda}_i$
4 **for** $j \in A_i$
5 $\boldsymbol{\Psi}_{ij} = -\boldsymbol{K}_{ii}^{-1}\boldsymbol{K}_{ij}$
6 **for** $k \in \widetilde{A}_j$
7 $\boldsymbol{M}_{jk} = \boldsymbol{M}_{jk} + \boldsymbol{\Psi}_{ij}^{T}\boldsymbol{M}_{ik} + \boldsymbol{M}_{ij}^{T}\boldsymbol{\Psi}_{ik} + \boldsymbol{\Psi}_{ij}^{T}\boldsymbol{M}_{ii}\boldsymbol{\Psi}_{ik}$
8 $\boldsymbol{K}_{jk} = \boldsymbol{K}_{jk} + \boldsymbol{\Psi}_{ij}^{T}\boldsymbol{K}_{ik}$
9 $\boldsymbol{M}_{ij} = \boldsymbol{\Phi}_i^{T}(\boldsymbol{M}_{ii}\boldsymbol{\Psi}_{ij} + \boldsymbol{M}_{ij})$
10 **for** $j \in D_i$
11 **for** $k \in A_i$
12 $\boldsymbol{M}_{jk} = \boldsymbol{M}_{jk} + \boldsymbol{M}_{ji}\boldsymbol{\Psi}_{ik}$
13 $\boldsymbol{M}_{ji} = \boldsymbol{M}_{ji}\boldsymbol{\Phi}_i$

After computing all nodes, the AMLS method assembles the matrices obtained from Algorithm 1 into a lower-dimensional generalized eigenvalue problem. By solving this problem, approximate values of the natural frequencies (eigenvalues) of the entire finite element model can be obtained. Finally, in the the back transformation stage, the eigenvectors of the original Eq. (1) are obtained.

Out-of-Core Implementation

In the AMLS method, each node generates substantial intermediate data during computation, such as the matrices Φ_i and Ψ_{ij}. It is noted that these matrices are not reused until the back transformation stage after node i has completed its computations. Given these considerations, transferring these intermediate data to disk is a reasonable approach.

Typically, the stiffness and mass matrices of the model are highly sparse, resulting in fewer coupling degrees of freedom (DOFs) between substructures. Before the model transformation stage, a map container is used to record these coupling relationships. All subsequent node calculations are based on this information. For instance, in practical calculations, the number of columns in the matrix Ψij in line 5 of Algorithm 1 is determined by the coupling DOFs between substructure i and substructure j, rather than the DOFs of substructure j. This greatly reduces the computational load.

The substructure solving process during the model transformation stage is illustrated in Figure 2. The same memory block is reused for matrix assembly and computation when calculating different substructures. Memory is only released and reallocated when the current memory is becomes insufficient. Experiments have shown that this approach helps mitigate memory fragmentation.

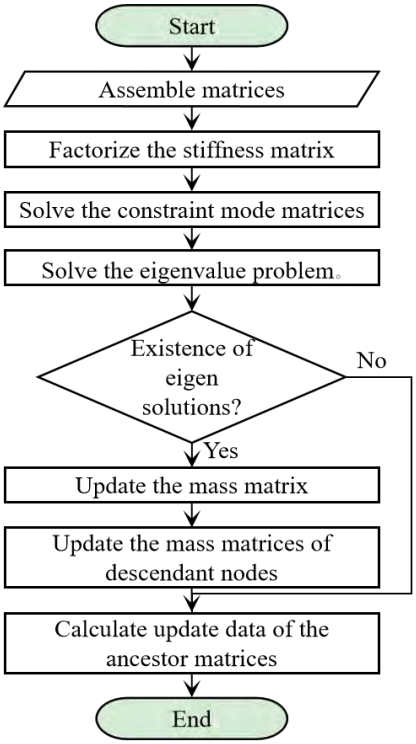


Figure 2. Substructure solving process during model transformation

In the node matrix assembly stage of the out-of-core program, different calculation strategies are applied to substructures at different positions in the substructure tree, as shown in Figure 3. All substructures are divided into three categories: leaf substructures (the leaf nodes in the substructure tree), next-to-bottom substructures (the parent substructures of the leaf substructures), and other substructures. The specific node assembly and computation process under the out-of-core method is illustrated in Figure 3.

Among these categories, the assembly process of leaf substructures is the simplest because they do not need to fetch update data from other substructures. All relevant matrix computations are performed using sparse matrix formats to leverage the sparsity of the matrices and improve computational efficiency. In our out-of-core parallel method, next-to-bottom substructures do not need to fetch update data from disk. This is due to two reasons: first, the update data from leaf substructures to their ancestor nodes is minimal; second, the update data from leaf substructures is quickly utilized by their parent nodes. For the third category of substructures, they need to perform three steps: assembling the original matrices, fetching data from disk, and updating the matrices. To implement these assembly schemes, it is necessary to transfer the update data of non-leaf nodes to disk. Figure 3(b) illustrates the types of data transferred to the disk and the specific timing during the node computation process.

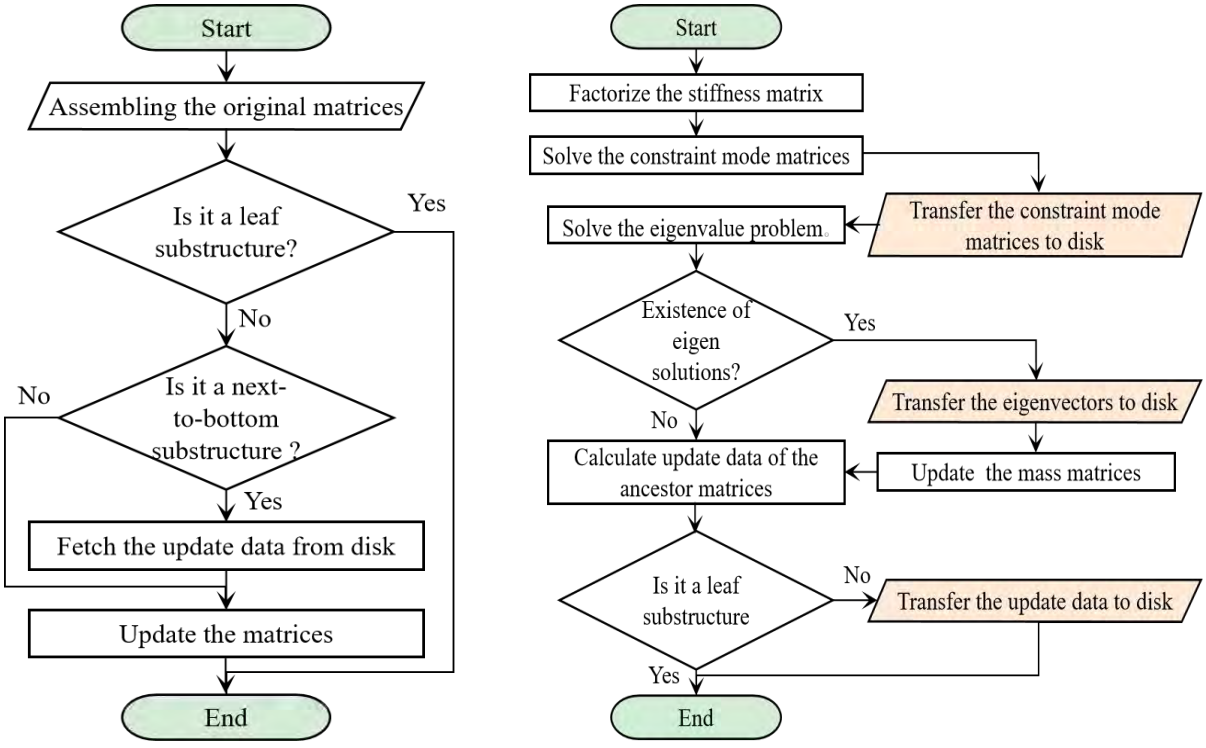


Figure 3. Substructure solving process in OOC mode: (a) Assembly, (b) Computation

Parallel Computing

Parallel computing on multiple cores is a key strategy to enhance the efficiency of the AMLS algorithm. In the AMLS method, various tasks of different granularities can be parallelized. For instance, computations among substructures at the same level in the substructure tree can be executed concurrently. Within an individual substructure, eigenvalue problems and constraint mode matrix calculations can also be parallelized. Several parallel AMLS methods are already available [9]-[10], and these strategies are equally viable in out-of-core computations. However, additional consideration are required for parallelizing data writes to disk.

From Figure 3, it is evident that we transfer three types of data to disk. These are categorized into two groups. We classify the eigenvectors and constraint mode matrices of substructures as the first group of data. These are only utilized during the back transformation stage after computations at the current node are completed. The second group consists of the update data of substructures to their ancestor nodes.

In our method, each thread creates a binary file on disk to store the first group of transfer data it computes. Each substructure records the filename and offset where its data is written. This approach ensures that the required data can be directly accessed during the back transformation stage without unnecessary thread conflicts.

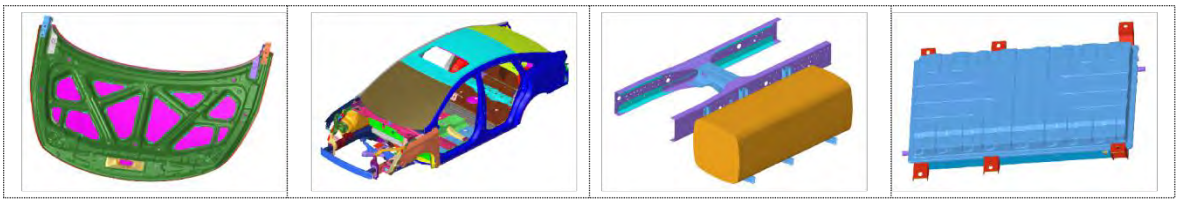
On the other hand, the second group of transfer data for each substructure is stored individually as a file on disk. These data are written, read, and deleted sequentially. The algorithm described in reference [10] guarantees their correct order.

Numerical experiments

To validate the effectiveness and efficiency of the proposed out-of-core automated multi-level substructuring method, several numerical experiments on engineering models were conducted. These experiments demonstrate that, compared to in-core (IC) AMLS methods, the OOC approach significantly reduces memory usage with only a slight loss in computational efficiency.

The algorithms discussed in this paper were implemented in C++ using OpenMP [14] for parallel processing. The testing environment specifications are as follows:

- CPU: 11th Gen Intel(R) Core(TM) i9-11900K @ 3.50GHz
- Memory: 128GB
- Disk: SSD Samsung MZVL21T0HCLR-00B00



(a) (b) (c) (d)

Figure 4. Finite Element Mode

We selected large-scale structural models representative of industrial applications as numerical examples, as shown in Figure 4. From left to right, these models are the car engine hood, the body-in-white, the fuel tank, and the battery pack. Each of these models represents critical components in their respective domains, chosen for their complexity and relevance.

Table 1. Finite element models for numerical experiments.

	Model	DOFs	Number of nonzero entries in the upper triangular part		Number of substructures
			Stiffness matrix	Mass matrix	
1	hod_233	2334655	63329790	2567104	7871
2	car_430	4308780	119676345	9349811	15997
3	tan_522	5221758	142570761	4623326	17001
4	bat_867	8676657	240060222	10477667	29171

Table 1 provides detailed information on these models, including the degrees of freedom, the number of non-zero elements in the matrices, and the number of substructures obtained through partitioning. This comprehensive data highlights the scale and complexity of each model, emphasizing the challenges involved in their computational analysis. By examining these parameters, we can better understand the computational demands and the effectiveness of our out-of-core parallel method in handling large-scale structural models.

We used both the out-of-core parallel method described in this paper and the in-core method to calculate the natural frequencies and mode shapes of these models in the 0-300Hz range. Figure 5 shows the resource consumption comparison between the two methods, with specific data presented in Table 2. Memory usage for storing eigenvectors is excluded from these calculations. The results indicate significant improvements in memory usage with the out-of-core method.

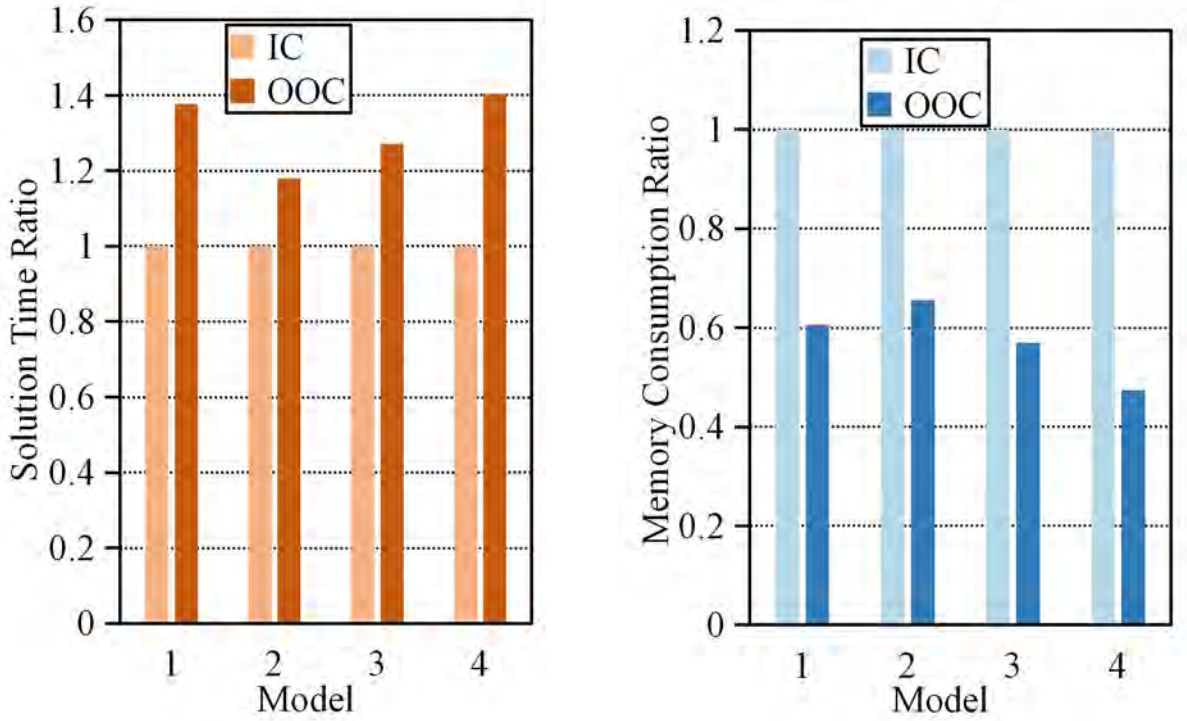


Figure 5. Resource consumption comparison between IC and OOC: (a) Solution Time, (b) Memory Consumption

Compared to the in-core method, the OOC method significantly reduces memory consumption while maintaining computational efficiency, demonstrating clear advantages. Specifically, the data show that for the large-scale structural models tested, the OOC method consistently used approximately 34% to 53% less memory. Although there is a slight increase in computational time with the OOC method, this trade-off is acceptable given the substantial memory savings. This indicates that the OOC method is effective for handling large-scale simulations on systems with limited memory capacity, making it a practical solution for such scenarios.

Table 2. Solution time and memory usages between IC and OOC

	Model	Times (sec)		Peak Memory Usage (GB)	
		In-core	Out-of-core	In-core	Out-of-core
1	hod_233	47.504	65.366	15.211	9.213
2	car_430	151.885	179.109	36.561	23.951
3	tan_522	147.411	187.506	40.930	23.326
4	bat_867	152.340	213.952	48.572	23.028

Conclusions

In this paper, we present an out-of-core parallel AMLS method to address the challenges of excessive memory consumption in large-scale structural dynamic analysis. By transferring intermediate data to disk and designing efficient data structures, our approach significantly reduces peak memory usage while maintaining computational efficiency. Several numerical experiments have validated this reduction in memory consumption. These findings suggest

that the out-of-core method is a practical solution for handling large-scale simulations on systems with limited memory capacity.

References

[1] BENNIGHOF, JEFFREY, and CONNIE KIM. "An adaptive multi-level substructuring method for efficient modelingof complex structures." 33rd Structures, Structural Dynamics and Materials Conference. 1992.
[2] Bourquin, Frédéric. "Component mode synthesis and eigenvalues of second order operators: discretization and algorithm." ESAIM: Mathematical Modelling and Numerical Analysis 26.3 (1992): 385-423.
[3] Bennighof, Jeffrey K., and Matthew F. Kaplan. "Frequency window implementation of adaptive multi-level substructuring." (1998): 409-418.
[4] Voß, Heinrich, and Kolja Elssel. "Automated Multilevel Substructuring for Nonlinear Eigenproblems." Preprints des Institutes für Mathematik (2005).
[5] Elssel Kolja, and Heinrich Voss. "Reducing huge gyroscopic eigenproblems by automated multi-level substructuring." Archive of Applied Mechanics 76 (2006): 171-179.
[6] Stammberger, Markus, and Heinrich Voss. "Automated multi-level sub-structuring for fluid–solid interaction problems." Numerical Linear Algebra with Applications 18.3 (2011): 411-427.
[7] Kaplan, Matthew Frederick. Implementation of automated multilevel substructuring for frequency response analysis of structures. The University of Texas at Austin, 2001.
[8] Kim, Jin-Gyun, Seung-Hwan Boo, and Phill-Seung Lee. "An enhanced AMLS method and its performance." Computer Methods in Applied Mechanics and Engineering 287 (2015): 90-111.
[9] Hyun, Cheolgyu, Seung-Hwan Boo, and Phill-Seung Lee. "Improving the computational efficiency of the enhanced AMLS method." Computers & Structures 228 (2020): 106158.
[10] Hyun, Cheolgyu, and Phill-Seung Lee. "A load balancing algorithm for the parallel automated multilevel substructuring method." Computers & Structures 257 (2021): 106649.
[11] Zhou, Zheng, et al. "An out-of-core eigensolver on SSD-equipped clusters." 2012 IEEE International Conference on Cluster Computing. IEEE, 2012.
[12] Gunter, Brian C., and Robert A. Van De Geijn. "Parallel out-of-core computation and updating of the QR factorization." ACM Transactions on Mathematical Software (TOMS) 31.1 (2005): 60-78.
[13] Karypis, George, and Vipin Kumar. "METIS: A software package for partitioning unstructured graphs." Partitioning Meshes, and Computing Fill-Reducing Orderings of Sparse Matrices, Version 4.0 (1998).
[14] Dagum, Leonardo, and Ramesh Menon. "OpenMP: an industry standard API for shared-memory programming." IEEE computational science and engineering 5.1 (1998): 46-55.

Mesoscale damage and fracture simulation of lightweight aggregate concrete under compression based on Voronoi-random walk algorithm

***Binhui Wang[1], †Xiaogang Song[1], Zihua Zhang[1, 2]**

1 Department of Civil Engineering, Ningbo University, Ningbo 315211, China

2 Key Laboratory of Impact and Safety Engineering (Ningbo University), Ministry of Education, Ningbo 315211, China

*Presenting author: 2211110038@nbu.edu.cn

†Corresponding author: songxiaogang@nbu.edu.cn

Abstract

The article investigates the damage and fracture development in lightweight aggregate concrete (LWAC) through mesoscale simulation. In the mesoscale model, the aggregates are generated by a hybrid algorithm combining Voronoi diagrams with the random walk algorithm (RWA) to achieve ideal gradation and a higher content of aggregate. The meso-structure of LWAC, composed of cement, aggregates, and interfacial transition zones (ITZs), was modeled by the proposed method. The crack propagation and damage evolution of LWAC under uniaxial compression were simulated with the finite element method. The numerical results agree well with the experimental observations, indicating the effectiveness and accuracy of the proposed approach.

Keywords: Mesostructure of concrete; Lightweight aggregate concrete; Crack propagation; Finite element analysis.

1 Introduction

Lightweight Aggregate Concrete (LWAC) finds extensive use in tall buildings, bridges, and underground passages due to its lighter weight and exceptional thermal insulation characteristics. The enclosed pores in the lightweight aggregates not only lessen the aggregate weight but also notably boost LWAC's thermal insulation capability. Nonetheless, owing to the comparatively lower strength of lightweight aggregates, the failure patterns of LWAC differ significantly from normal concrete. Typically, the failure of LWAC happens not just in the interface transition zones (ITZs) or the mortar matrix, but also potentially in the lightweight aggregate particles themselves, commonly through compression or tension. Thus, it is imperative to investigate and enhance the mechanical properties of LWAC while maintaining its lightweight features.

Among the mesoscale methodologies related to LWAC, two-dimensional (2D) models are predominantly used. Jin et al. [1,2] conducted numerical simulations on circular LWAC, modeling the aggregate particles, mortar matrix, and ITZs in a 2D plane, incorporating a two-level distribution, and describing the failure behavior of lightweight aggregate particles using a plastic damage constitutive model. Wang et al. [3] developed a 2D LWAC model,

discretizing the lightweight aggregate particles and inserting zero-thickness cohesive elements to simulate their fracture. The above-mentioned 2D models overlook all geometric features and structural changes in the third dimension, leading to deviations from realistic LWAC. Three-dimensional (3D) mesoscale models can more accurately reflect the random distribution of aggregates and their state under different loading conditions. Chen et al. [4] established a coral aggregate concrete (CAC) mesoscale model composed of aggregates and mortar matrix, in which aggregate particles and mortar were discretized and cohesive elements were inserted into neighboring elements to study the damage evolution and fracture modes of CAC. Wu et al. [5] established a CAC mesoscale model consisting of coral aggregate, mortar matrix, and ITZs, and the K & C constitutive model [6] was utilized to describe the mechanical properties of CAC under uniaxial and multi-axial loading.

This study investigates the mechanical properties and failure behavior of shale ceramsite LWAC under uniaxial compression by a mesoscale model. A self-developed Voronoi-random walk algorithm, valued for its quick placement speed and high aggregate content, is employed for aggregate placement. The lightweight aggregates are represented using four-node tetrahedral elements, and their failure mechanisms are captured through a damage plasticity model. Furthermore, the ITZs between the aggregates and the mortar matrix are simulated using six-node wedge cohesive elements, and their failure characteristics are defined by a bilinear constitutive model.

2 Finite element modeling

2.1 *3D mesostructure of concrete*

2.1.1 *Aggregate model*

Based on seed points, Voronoi tessellation divides the target domain into several subregions, ensuring that any point within a subregion is closer to its respective seed point than to any other seed points. Caballero et al. [7] pioneered the use of Voronoi diagrams for generating aggregates in concrete. In this study, the Random Walk Algorithm (RWA) [8] is utilized to position seed points and achieve well-distributed Voronoi polyhedra. New geometric optimization and gradation adjustment algorithms are introduced to improve the quality of the polyhedra and modify the aggregate gradation distribution.

A target domain of 50 mm × 50 mm × 50 mm is used to generate the aggregate model, with the particle size distribution shown in Table 1. A four-level gradation of aggregates is adopted. leading to an aggregate content of 30.33%.

Table 1. Particle size distribution of aggregates

Sieve size (mm)	Total percentage retained (%)	Total percentage passing (%)
19.0	0	100
12.7	19.90	80.10
9.5	41.52	58.48
4.75	55.27	44.73
2.36	100	0

2.1.2 *Mortar and ITZ model*

The analysis of mortars and ITZs is carried out in ABAQUS. Aggregate shapes and spatial distributions are input into ABAQUS for modeling, where the mortar region is obtained by Boolean operations. Previous studies [3,9–13] have shown that zero-thickness cohesive elements effectively simulate ITZ damage characteristics with minimal mesh elements. In this study, we utilize these cohesive elements to represent ITZs at the lightweight aggregate-mortar interfaces with a mesostructure illustrated in Fig. 1. C3D4 elements were used for mortar and aggregates, while COH3D6 elements were used for internal interfaces.

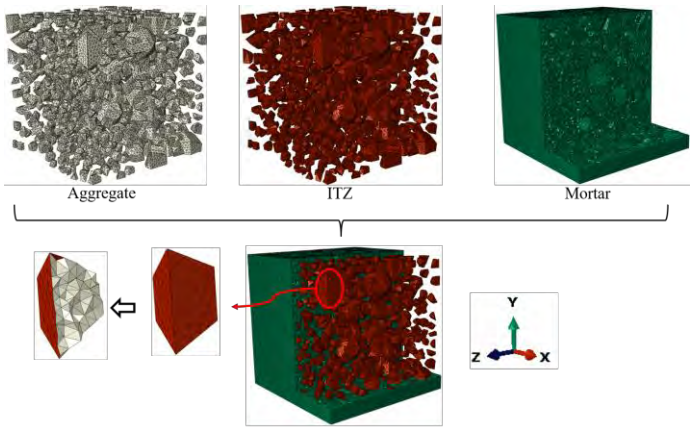


Figure 1. Mesostructure of LWAC

2.2 *Material parameters*

The plastic damage constitutive model proposed by Lee and Fenves [14] described the mechanical behavior of aggregate. Previous studies[2,9,15–18] have validated the Concrete Damage Plasticity (CDP) model as an effective tool for simulating the mechanical responses of mortar [9,16–18] and lightweight aggregates [2,15]. This model includes compressive strain hardening and softening curves as per GB50010 standards [19], with its constitutive relationship depicted in Fig. 2. Damage extent in the model is quantified by "SDEG", a scalar stiffness degradation variable ranging from 0 (undamaged state) to 1 (complete damage) [10]. Cracked elements are those with an SDEG value exceeding 0.9 [9,17]. In previous studies [3,10,12], ITZs are modeled using zero-thickness cohesive elements. The bilinear constitutive law, known for its computational efficiency and accuracy [12], is chosen in this research. The quadratic nominal stress criterion and mixed-mode fracture criterion [11] are illustrated in Fig. 3. The main material properties are shown in Table 2.

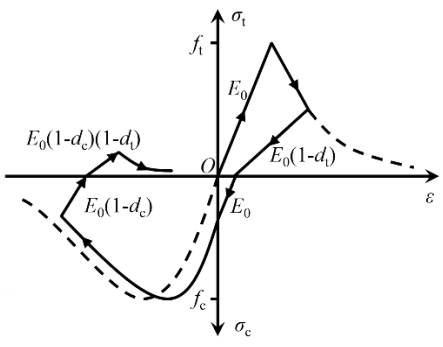


Figure 2. Concrete damaged plasticity model

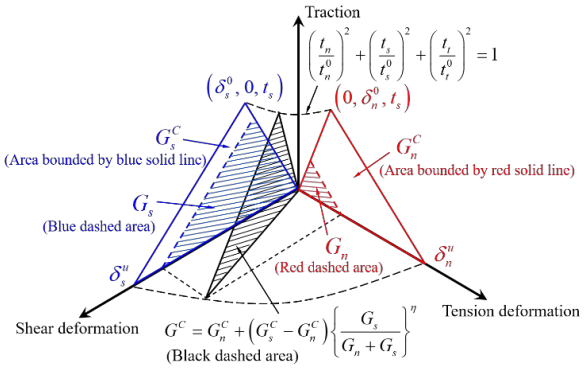


Figure 3. Quadratic nominal stress criterion and mixed-mode fracture criterion

Table. 2 Material properties of the mesoscale model

Material properties	Aggregate	Mortar	ITZ
Density, ρ (kg/m^3)	1100	2120	2000
Elasticity modulus, E (GPa)	10	29.4	-
Poisson rate, v	0.22	0.2	-
Compressive strength, f_c (MPa)	11	45.8	-
Tensile strength, f_t (MPa)	2.7	4.5	-
Fracture energy, G_f (N/mm)	0.02	0.06	-
Cohesive stiffness, E/E_{nn}, G_1/E_{ss} and G_2/E_{tt} (MPa/mm)	-	-	10^6
Maximum nominal stress in the normal direction, t_n^0 (MPa)	-	-	2.6
Maximum nominal stress in shear direction, t_s^0 and t_t^0 (MPa)	-	-	10
Normal mode fracture energy, G_n^C (N/mm)	-	-	0.025
Shear mode fracture energy, G_s^C (N/mm)	-	-	0.0625

2.3 *Loads and boundary conditions*

In simulating the uniaxial compression of a concrete cubic specimen, a finite element analysis is performed using the dynamic explicit method. The loading is applied as a displacement with two pads placed along the Y-axis of the specimen. The upper pad moves vertically by 0.2 mm in the negative Y-direction, while the lower pad stays stationary. Guo et al. [20] suggested that a friction coefficient in the range of 0.1 to 0.7 can reflect realistic experimental conditions. Zhang et al. [21] confirmed that the failure mode of concrete at a coefficient of friction of 0.6 was similar to the experiment. Therefore, the friction coefficients between the upper and lower pads and the specimen are set as 0.6.

3 Model reliability

All numerical models are simulated using ABAQUS on an AMD EPYC 7542 32-Core Processor @ 3.4 GHz desktop computer with 128 GB of RAM.

3.1 *Mesh sensitivity*

In nonlinear finite element analysis, the size of the element mesh significantly affects computational precision. Choosing the right mesh size is crucial for balancing accuracy and efficiency. After running multiple simulations, we discovered that mesh sensitivity is reduced when the cell size is below 1 mm. Thus, we select 1 mm as the element size.

3.2 *Loading rate*

In dynamic explicit quasi-static simulations, energy changes are utilized to evaluate the precision of the simulation outcomes. If the total kinetic energy to total internal energy ratio is under 5%, inertial effects can be ignored, indicating precise results [22]. To ensure dependable simulation results, an appropriate loading rate must be established. The simulation revealed that the stress-strain curve of the LWAC model remains consistent when the loading rate is below 20 mm/s. The ratio of kinetic energy to internal energy also complies with the set criteria, hence 20 mm/s is selected as the loading rate.

4 Results and discussion

4.1 *Comparison of simulations with experiments*

The failure mode of a specified specimen obtained by experiment is demonstrated in Fig. 4(a). In Fig. 4(b), damaged cross-sections of aggregates are represented by white, undamaged aggregates encapsulated by ITZs are indicated by red, and the mortar matrix is denoted by green. Fig. 4(c) illustrates the final damage contour of the FE model. Stress-strain curves of the experiment and simulation are shown in Fig. 5, where Test 1 and Test 2 correspond to the experimental results with a 30% lightweight aggregate content.

The curves are analyzed in stages: Before point A, the elastic phase is observed. At point B, the initial crack emerges in the damage cloud diagram, with a steep slope in the preceding descending section, likely attributed to crack propagation through aggregates and ITZs. After point B, a rise in the simulated curve is noted as the remaining lightweight aggregates bear the load, leading to mortar damage around these aggregates and a slight stress increase. Further strain results in a sharp stress drop between points B and C, accompanied by a second crack at point C in the damage cloud diagram. Multiple lightweight aggregate particles intersect the crack, while ITZs fractures escalate. The curve rises again after point C, showing a significant stress rise, possibly due to smaller lightweight aggregate particles approaching failure limits and larger aggregates begining to be damaged, as supported by the damage diagram at point D. As compression advances towards failure, extensive mortar damage occurs, and more lightweight aggregate particles are fractured or crushed, as depicted in the final damage state at point E. Compared to the experimental curves, the simulated curve closely aligns before point B. Experimental curve 1 displays slope changes at strains of 0.0018 and 0.0024 in the descending section, similar to the simulated curve. However, beyond these points, experimental curves do not exhibit an upward trend, likely due to the brittleness of real lightweight aggregates, with cracks spreading along the ITZ, rapidly intersecting lightweight aggregate particles, and extending into the mortar matrix. The sharp decline in experimental curve 2 occurs later, around a strain of 0.0021, followed by a plateau. The random distribution of aggregates in the mortar leads to the phenomenon. The peak load difference between the simulated and experimental curves is minimal overall, with a generally consistent trend, indicating that the numerical model can effectively replicate experiments, providing a solid technical basis for subsequent numerical analyses and reducing the need for further experimental work.

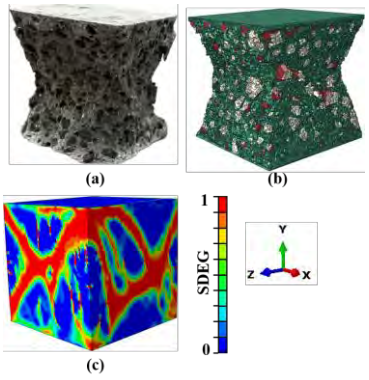


Figure 4. (a) Failure mode of the experimental sample, (b) Simulated failure mode, and (c) SDEG (scalar stiffness degradation variable) of the concrete model

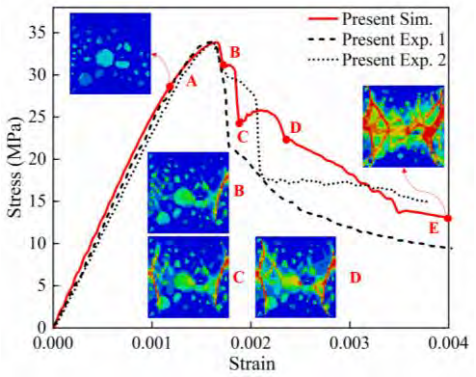


Figure 5. Stress-strain curves of experiments and simulations

4.2 *Parameter analysis*

4.2.1 *Aggregate content*

Aggregate models with different contents 35%, 30%, and 25% are compared. The peak compressive load and elasticity modulus of LWAC are inversely proportional to the aggregate content, as shown in Fig. 6. As the LWA content decreases, the stress-strain curve of LWAC becomes more similar to that of the mortar matrix. At 25% LWA content, the stress-strain curve exhibits a smoother decreasing part without sudden drops, attributed to cracks propagating through the mortar and damaging the lightweight aggregate. With a mortar LWA content of 35%, slight fluctuations before reaching the peak load indicate damage to some lightweight aggregate, affecting the behavior in the falling part of the curve due to the random distribution of aggregates.

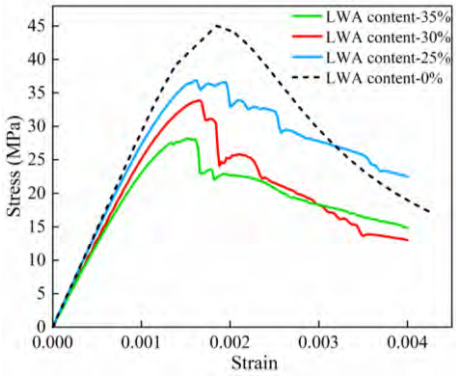


Figure 6. Stress-strain curves of different samples with various LWA contents

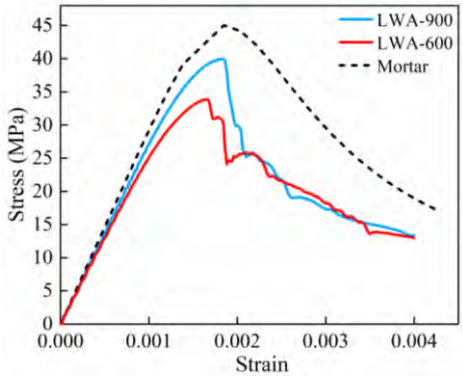


Figure 7. Stress-strain curves of different samples with various LWA grades

4.2.2 *Strength of aggregate*

To consider the impact of LWA strength grades on the stress-strain behavior of LWAC, mesoscale concrete models have been developed. The characteristics of the mortar and LWA 600 are shown in Table 2. Specifically, for LWA grade 900, the compressive strength (f_{ac}) is 23.37 MPa, the tensile strength (f_{at}) is 4.8 MPa, and the modulus of elasticity is 22 GPa. In Fig. 7, the stress-strain curves for concrete models incorporating two distinct LWA grades are depicted. A comparison reveals that the LWA-900 curve demonstrates a higher elasticity modulus and increased strength when compared to the LWA-600 curve. Notably, the stress-strain behavior of LWAC with LWA grade 900 closely resembles that of pure mortar, indicating that as the aggregate strength improves, the mechanical properties of the corresponding LWAC tend to imitate those of mortar.

5 Conclusions

In this study, a self-developed aggregate placement algorithm was used to generate a three-dimensional polyhedral aggregate model for simulating shale ceramsite lightweight aggregate concrete. Some remarkable conclusions can be achieved:

(1) The study focused on analyzing the damage and fracture patterns of LWAC under uniaxial compression. The stress-strain curves and ultimate failure modes showed that the proposed approach can simulate the experimental observations effectively. Through monitoring damage progression within the concrete model at the mesoscale, the primary failure mechanism of

LWAC was identified, starting from the weaker parts of ITZs. With increased loading, cracks propagated through smaller LWA particles, merging into larger cracks. Eventually, these cracks led to the failure of LWAC, displaying an hourglass-shaped failure pattern.

(2) The mechanical properties of LWAC under uniaxial compression are notably affected by aggregate content and strength, showing a complex relationship. Peak compressive load and elasticity modulus exhibit an inverse correlation with aggregate content. LWAC with lower LWA content exhibits strengths more similar to pure mortar. Moreover, utilizing higher strength grade lightweight aggregates enhances both the elasticity modulus and strength of LWAC, aligning its strength more closely with pure mortar.

References

[1] Jin, L., Yang, W., Yu, W. and Du, X. (2021) Study on Failure Behavior and Size Effect of Lightweight Aggregate Concrete Based on Meso-Simulation. *Journal of Disaster Prevention and Mitigation Engineering*, **41**, 91–99. https://doi.org/10.13409/j.cnki.jdpme.2021.01.011.

[2] Jin, L., Yu, W., Du, X., Zhang, S. and Li, D. (2019) Meso-Scale Modelling of the Size Effect on Dynamic Compressive Failure of Concrete under Different Strain Rates. *International Journal of Impact Engineering*, **125**, 1–12. https://doi.org/10.1016/j.ijimpeng.2018.10.011.

[3] Wang, B., Zhu, E. and Zhang, Z. (2022) Microscale Fracture Damage Analysis of Lightweight Aggregate Concrete under Tension and Compression Based on Cohesive Zone Model. *Journal of Engineering Mechanics*, American Society of Civil Engineers, **148**, 04021153. https://doi.org/10.1061/(ASCE)EM.1943-7889.0002051.

[4] Chen, B., Yu, H., Zhang, J. and Ma, H. (2022) Evolution Law of Crack Propagation and Crack Mode in Coral Aggregate Concrete under Compression: Experimental Study and 3D Mesoscopic Analysis. *Theoretical and Applied Fracture Mechanics*, **122**, 103663. https://doi.org/10.1016/j.tafmec.2022.103663.

[5] Wu, Z., Yu, H., Zhang, J. and Ma, H. (2023) Mesoscopic Study of the Mechanical Properties of Coral Aggregate Concrete under Complex Loads. *Composite Structures*, **308**, 116712. https://doi.org/10.1016/j.compstruct.2023.116712.

[6] Wu, Z., Zhang, J., Yu, H., Ma, H., Chen, L., Dong, W., Huan, Y. and Zhang, Y. (2020) Coupling Effect of Strain Rate and Specimen Size on the Compressive Properties of Coral Aggregate Concrete: A 3D Mesoscopic Study. *Composites Part B: Engineering*, **200**, 108299. https://doi.org/10.1016/j.compositesb.2020.108299.

[7] Caballero, A., López, C.M. and Carol, I. (2006) 3D Meso-Structural Analysis of Concrete Specimens under Uniaxial Tension. *Computer Methods in Applied Mechanics and Engineering*, **195**, 7182–7195. https://doi.org/10.1016/j.cma.2005.05.052.

[8] Zhang, Z., Song, X., Liu, Y., Wu, D. and Song, C. (2017) Three-Dimensional Mesoscale Modelling of Concrete Composites by Using Random Walking Algorithm. *Composites Science and Technology*, **149**, 235–245. https://doi.org/10.1016/j.compscitech.2017.06.015.

[9] Zhou, G. and Xu, Z. (2023) 3D Mesoscale Investigation on the Compressive Fracture of Concrete with Different Aggregate Shapes and Interface Transition Zones. *Construction and Building Materials*, **393**, 132111. https://doi.org/10.1016/j.conbuildmat.2023.132111.

[10] Zhang, L., Sun, X., Xie, H. and Feng, J. (2023) Three-Dimensional Mesoscale Modeling and Failure Mechanism of Concrete with Four-Phase. *Journal of Building Engineering*, **64**, 105693. https://doi.org/10.1016/j.jobe.2022.105693.

[11] Xiong, X. and Xiao, Q. (2019) Meso-Scale Simulation of Concrete Based on Fracture and Interaction Behavior. *Applied Sciences*, Multidisciplinary Digital Publishing Institute, **9**, 2986. https://doi.org/10.3390/app9152986.

[12] Wang, X., Zhao, T., Guo, J., Zhang, Z. and Song, X. (2023) Mesoscale Modelling of the FRP-Concrete Debonding Mechanism in the Pull-off Test. *Composite Structures*, **309**, 116726. https://doi.org/10.1016/j.compstruct.2023.116726.

[13] Wang, Z.M., Kwan, A.K.H. and Chan, H.C. (1999) Mesoscopic Study of Concrete I: Generation of Random Aggregate Structure and Finite Element Mesh. *Computers and Structures*, **70**, 533–544. https://doi.org/10.1016/S0045-7949(98)00177-1.

[14] Lee, J. and Fenves, G.L. (1998) Plastic-Damage Model for Cyclic Loading of Concrete Structures. *Journal of Engineering Mechanics*, American Society of Civil Engineers, **124**, 892–900. https://doi.org/10.1061/(ASCE)0733-9399(1998)124:8(892).

[15] Wang, X., Liu, L., Zhou, H., Song, T., Qiao, Q. and Zhang, H. (2021) Improving the Compressive Performance of Foam Concrete with Ceramsite: Experimental and Meso-Scale Numerical Investigation. *Materials & Design*, **208**, 109938. https://doi.org/10.1016/j.matdes.2021.109938.

[16] Zheng, Y., Zhang, Y., Zhuo, J., Zhang, P. and Hu, S. (2023) Mesoscale Synergistic Effect Mechanism of Aggregate Grading and Specimen Size on Compressive Strength of Concrete with Large Aggregate Size. *Construction and Building Materials*, **367**, 130346. https://doi.org/10.1016/j.conbuildmat.2023.130346.

[17] Naderi, S., Tu, W. and Zhang, M. (2021) Meso-Scale Modelling of Compressive Fracture in Concrete with Irregularly Shaped Aggregates. *Cement and Concrete Research*, **140**, 106317. https://doi.org/10.1016/j.cemconres.2020.106317.

[18] Ma, D., Liu, C., Zhu, H., Liu, Y., Jiang, Z., Liu, Z., Zhou, L. and Tang, L. (2022) High Fidelity 3D Mesoscale Modeling of Concrete with Ultrahigh Volume Fraction of Irregular Shaped Aggregate. *Composite Structures*, **291**, 115600. https://doi.org/10.1016/j.compstruct.2022.115600.

[19] Du, X., Jin, L. and Ma, G. (2014) Numerical Simulation of Dynamic Tensile-Failure of Concrete at Meso-Scale. *International Journal of Impact Engineering*, **66**, 5–17. https://doi.org/10.1016/j.ijimpeng.2013.12.005.

[20] Guo, Y.B., Gao, G.F., Jing, L. and Shim, V.P.W. (2017) Response of High-Strength Concrete to Dynamic Compressive Loading. *International Journal of Impact Engineering*, **108**, 114–135. https://doi.org/10.1016/j.ijimpeng.2017.04.015.

[21] Zhang, Y., Wang, Z., Zhang, J., Zhou, F., Wang, Z. and Li, Z. (2019) Validation and Investigation on the Mechanical Behavior of Concrete Using a Novel 3D Mesoscale Method. *Materials (Basel, Switzerland)*, **12**, 2647. https://doi.org/10.3390/ma12162647.

[22] Gao, S. (2019) Nonlinear Finite Element Failure Analysis of Bolted Steel-Concrete Composite Frame under Column-Loss. *Journal of Constructional Steel Research*, **155**, 62–76. https://doi.org/10.1016/j.jcsr.2018.12.020.

Modeling, simulation and analysis of caving process of multi-layered loose top-coal based on the SPH method

***†Liu Yang[1], Zhang Qiang[1], and Dong Xiangwei[1]**

[1]Collage of Mechanical and Electronic Engineering, Shandong University of Science and Technology, Qianwangang Road 579, Qingdao 266590, China

*Presenting author: 2493470410@qq.com
†Corresponding author: 2493470410@qq.com

Abstract

To study the caving characteristics of top coal in longwall mining, a mesh-free numerical model based on Smoothed Particle Hydrodynamics (SPH) was proposed. Initially, SPH discrete equations for the governing equations of continuum mechanics were developed, incorporating an elastoplastic soil constitutive model and the Drucker-Prager yield criterion to simulate the collapse, migration, and release processes of granular top coal. Subsequently, considering the actual coal discharge and transportation processes in the mining site, a scraper conveyor model was constructed to simulate the horizontal top coal release and bottom coal transportation along the working face. The variations in coal-rock interface and coal flow velocity were analyzed under different scraper conveyor speeds (0.0-1.5 m/s). The results indicate that the non-cohesive soil constitutive model effectively simulates the flow behavior of granules by setting material parameters such as friction angle and modulus, thereby avoiding the parameter uncertainty issue in traditional Discrete Element Method (DEM) models. After the coal flow velocity stabilizes, the stress distribution near the coal discharge opening exhibits a "double peak" pattern. The scraper conveyor speed significantly influences the coal discharge time but has minimal impact on the final coal-rock interface and discharge body shape. When multiple supports simultaneously discharge coal, the conveying capacity of the scraper conveyor must be considered, as interference in bottom coal transportation between different supports can lead to blockage at the discharge openings. The "close when gangue appears" criterion results in discharge quantity variations among different openings, with the standard deviation of top coal release from 40 discharge openings being 7.52 m², higher than the standard deviation of 1.93 m² for automatic coal discharge.

Keywords: SPH, granular coal particles, coal discharge-transportation coupling, coal-rock interface, coal flow velocity

1. Introduction

In China, thick coal seams account for 45% of the underground coal resources and production. The top coal caving method is one of the main techniques for mining thick coal seams[1]. Top coal caving technology has been undergoing industrial trials and applications in China for over 30 years, gradually entering a mature stage of independent research and innovation[2]. The traditional top coal caving technology requires manual control of the coal discharge openings, with the termination condition being "close when gangue appears." Currently, top coal caving has started to evolve towards intelligent mining, utilizing automated top coal caving control systems to achieve mechanized top coal caving in longwall faces[3]. Due to the complexity of the top coal caving process and conditions, many challenges remain to be overcome to fully realize intelligent longwall top coal caving[4]. Previous scholars have

directly assumed the top coal to be a loose body[5-6], studying the caving, migration, and arching mechanisms of granular materials, which is suitable for soft, friable, or fissured coal seams[7]. However, most previous studies have neglected the synergistic process between the scraper conveyor and the support system during coal discharge. In actual coal discharge processes, the scraper conveyor is a critical component ensuring continuous coal discharge and transportation. Therefore, it is necessary to consider the mutual influence between the support system's coal discharge and the scraper conveyor's coal transportation. Further establishing optimization methods for top coal caving processes that align with actual mining conditions will provide guidance for improving top coal recovery rates and discharge efficiency.

Numerical simulation is a crucial tool for studying the movement and release patterns of top coal. Common numerical methods include the Discrete Element Method (DEM), the Finite Difference Method (FDM), and the Finite Element Method (FEM). Representative software for DEM includes PFC and EDEM, while FDM and FEM are represented by software such as FLAC3D and LS-DYNA. Zhang et al.[8] used PFC2D software to simulate the coal-gangue flow field under different top coal thickness conditions based on the actual conditions of Hebi Ten Mine, analyzing the effects of coal seam thickness variations and different mining-to-caving ratios on the characteristics of the coal-gangue flow field and the release patterns of top coal. Wang et al.[9] employed PFC3D software to simulate the three-dimensional release morphology of coal and gangue during longwall top coal caving, analyzing the impact of different coal discharge methods on the spatial morphology of the release body. Sun et al.[10] studied the release patterns of coal in shallow-buried, hard, thick coal seams, using FLAC3D software to analyze the failure patterns of coal and rock during the longwall mining process. He et al.[11] using the 34201 working face of the Pingshuo Group as a background and combining it with on-site geological conditions, employed PFC2D to establish numerical models with dip angles of 0°, 13°, 23°, and 33°, analyzing the top coal displacement field, contact stress field, coal-gangue interface, initial displacement points of top coal, and the morphology of the release body under different conditions. Zhang et al.[12] based on the BBR research system, used PFC software to establish a numerical model, simulating the top coal release process under working face inclinations from 0° to 50°. They analyzed the impact of working face inclination on the granular top coal release patterns. Feng et al.[13] utilized LS-DYNA3D software to simulate the effect of different charging methods and blast hole positions on the fragmentation of the roof. Zou et al.[14] used PFC software to simulate the coal-rock interface under different coal seam inclinations, establishing a parabolic equation for the coal-rock interface equivalent to the release ellipsoid area. Jiang et al.[15] targeting the steeply inclined longwall face of Tangshan Mine, employed PFC2D software to study the asymmetric distribution characteristics of the coal-rock interface and optimize the coal discharge process. Wang et al.[16] used PFC2D software to study the movement and evolution patterns of granular top coal and roof in the re-mining face. Liu et al.[17] simulated and compared the top coal recovery rates and mining equilibrium under different caving techniques using PFC2D software, concluding that the independent cluster caving technique can improve the top coal recovery rate in thick coal seams. Wei et al.[18] measured the size distribution of top coal blocks and, through discrete element calculations and physical experiments, investigated the mixed movement mechanisms of top coal and gangue blocks in ultra-thick coal seams.

In longwall top coal caving (LTCC), the progressive failure process of top coal represents a typical continuous-discontinuous problem. Traditional methods such as the Discrete Element Method (DEM) and the Finite Element Method (FEM) are generally suitable for either

continuous or discontinuous problems but not both. Studying the continuous-discontinuous transition process requires coupling different methods. Zhang et al.[19] used FLAC3D and PFC3D software to achieve dynamic coupling simulation of continuous blocks and discrete particles through an interface coupling method, studying the progressive failure behavior of top coal in high-seam LTCC. Zhang et al.[20] adopted the Continuum-Discrete Element Method (CDEM) to simulate the top coal release process, incorporating the constitutive model of hydraulic supports to simulate the coupling interaction between hydraulic supports and surrounding rocks. Li et al.[21] used the CDEM method to establish a two-dimensional numerical model along the strike of the working face, exploring the impact of coal discharge step length on top coal loss and recovery rates. Liu et al.[22] employed the CDEM method to establish a numerical model along the dip of the working face, simulating the coal discharge process under different top coal thicknesses (4.0m, 8.0m, 12.0m) and analyzing the variation patterns of the coal-gangue interface and top coal recovery rates. However, the continuous-discontinuous coupling method involves complex coupling algorithms and must address the precise interaction of interface information during the transition process. In recent years, continuous particle methods, such as Smoothed Particle Hydrodynamics (SPH)[23] and the Material Point Method (MPM)[24] have developed rapidly. These methods not only inherit the advantages of mesh-free techniques in handling large deformation problems but also possess the deterministic characteristics of continuum mechanics models, providing a unified solution for continuous-discontinuous problems. Additionally, previous simulations of the coal discharge process often overlooked the transportation process of the scraper conveyor, assuming that the coal flow moves into unobstructed free space, which does not align with actual conditions.

This paper presents a numerical model of the top coal release and bottom coal transportation processes in longwall top coal caving using the Smoothed Particle Hydrodynamics (SPH) method, combined with a non-cohesive elastoplastic soil constitutive model to simulate the flow behavior of granular coal particles. SPH is a "continuous particle method," with its model based on continuum mechanics equations and material parameters possessing deterministic characteristics. Using virtual particle technology, a scraper conveyor model was constructed to simulate the bottom coal transportation process. By simulating the coal release and transportation processes, the evolution of the coal-rock interface, coal flow velocity, and the interaction between support coal release and scraper conveyor transportation are studied. The structure of this paper is as follows: Chapter 1 briefly introduces the numerical approximation principles of the SPH method; Chapter 2 establishes the SPH numerical discrete equations for the geotechnical control equations and introduces the implementation methods of the elastoplastic constitutive model and yield criterion; Chapter 3 presents the validation of the SPH numerical model and discusses the SPH simulation results of the coal release and transportation processes along the working face.

2. SPH Basic Theory

SPH is a numerical computation method that discretizes a continuous medium into "particles" and approximates physical quantities locally within an integration domain using kernel functions, thus simulating physical phenomena. In the SPH method, field functions are approximated using a "kernel integration" approach, where the field value and its derivatives are discretized by calculating the weighted sum of the field values of neighboring particles within a local integration domain. The construction of SPH discrete equations mainly involves two steps: first, the "kernel approximation," and second, the "particle approximation." The latter estimates the variable information of the current particle by

calculating the weighted sum of physical quantities of neighboring particles within the influence domain of the kernel function.

Based on the kernel approximation principle, any field function $f(\boldsymbol{x})$ can be approximated as follows:

$$f(\boldsymbol{x}) = \int_{\Omega} f(\boldsymbol{x}')W(\boldsymbol{x}-\boldsymbol{x}',h)dx' \tag{1}$$

In the equation, $f(x)$ represents any field function within the computational domain, $W(\boldsymbol{x}-\boldsymbol{x}',h)$ denotes the kernel function, Ω represents the integration domain or influence domain of the kernel function, and h is the smoothing length, which in this study is taken to be 1.2 times the initial particle spacing.

The kernel approximation expression for the gradient of a field function is as follows:

$$\nabla f(\boldsymbol{x}) = -\int_{\Omega} f(\boldsymbol{x}')\nabla W(\boldsymbol{x}-\boldsymbol{x}',h)dx' \tag{2}$$

In the equation, $\nabla W(\boldsymbol{x}-\boldsymbol{x}',h)$ represents the gradient of the kernel function.

Equation (2) represents the SPH kernel approximation in a continuous integral form. Since the computational domain is discretized into particles, as shown in Figure 1, the kernel approximation can be further transformed into a discrete form based on the distribution of particles, known as "particle approximation." It uses the superposition of supporting domain particles to complete the sum.

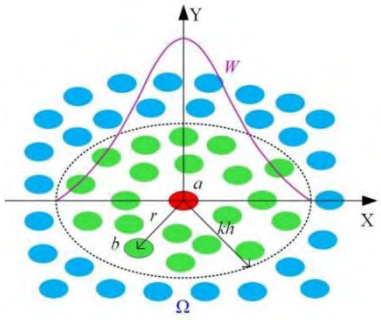


Fig 1. Support or influence domain of SPH particle i

Equations (1) and (2) can be written in the following discrete form, i.e., the particle summation approximation expression[23]:

$$\langle f(\boldsymbol{x}_i)\rangle = \sum_{j=1}^{N_i} \frac{m_j}{\rho_j} f(\boldsymbol{x}_j)W_{ij} \tag{3}$$

$$\langle \nabla f(\boldsymbol{x}_i)\rangle = -\sum_{j=1}^{N_i} \frac{m_j}{\rho_j} f(\boldsymbol{x}_j)\nabla_i W_{ij} \tag{4}$$

In the equations, $W_{ij} = W\left(\left|\boldsymbol{x}_i - \boldsymbol{x}_j\right|, h_{ij}\right)$ and $\nabla_i W_{ij} = \nabla_i W\left(\left|\boldsymbol{x}_i - \boldsymbol{x}_j\right|, h_{ij}\right) = \frac{\boldsymbol{x}_i - \boldsymbol{x}_j}{\left|\boldsymbol{x}_i - \boldsymbol{x}_j\right|}\frac{\partial W_{ij}}{\partial\left|\boldsymbol{x}_i - \boldsymbol{x}_j\right|}$, i represents the index of the particle, and j specifically refers to the index of the particle located within the support domain of particle i. The volume of particle j is $V_j = m_j / \rho_j$, where m_j is the mass and ρ_j is the density of the particle. N_i represents the total number of particles within the support domain of particle i.

When constructing the SPH discretized equations, the partial differential terms in the governing equations can be represented using the particle summation approximation (Equation 4), but some necessary transformations are required. Taking the divergence of a vector function as an example, in SPH, the divergence is typically approximated as follows[23]:

$$\langle \nabla \cdot f(\boldsymbol{x}_i) \rangle = \frac{1}{\rho_i} \left[\sum_{j=1}^{N_i} m_j \left(f(\boldsymbol{x}_j) - f(\boldsymbol{x}_i) \right) \cdot \nabla_i W_{ij} \right] \tag{5}$$

Here, $\langle \nabla \cdot f(\boldsymbol{x}_i) \rangle$ represents the particle approximation of the divergence of the vector function fff. Note that the kernel function is a function that decreases as the kernel distance increases, ensuring interactions between each particle and its neighboring particles within a certain truncation radius.

The particle approximation of the gradient of a scalar function can be expressed by the following two equations[23]:

$$\langle \nabla f(\boldsymbol{x}_i) \rangle = \rho_i \left[\sum_{j=1}^{N_i} m_j \left(\frac{f(\boldsymbol{x}_j)}{\rho_j^2} + \frac{f(\boldsymbol{x}_i)}{\rho_i^2} \right) \nabla_i W_{ij} \right] \tag{6}$$

$$\langle \nabla f(\boldsymbol{x}_i) \rangle = \rho_i \left[\sum_{j=1}^{N_i} m_j \left(\frac{f(\boldsymbol{x}_j) + f(\boldsymbol{x}_i)}{\rho_i \rho_j} \right) \nabla_i W_{ij} \right] \tag{7}$$

The equation (7) is used in this paper to discretize the gradient term in the control equation. It is worth noting that there is no significant difference in numerical approximation accuracy between equation (6) and equation (7). Based on the experience of this paper, equation (6) is also feasible.

The paper adopts a cubic spline piecewise function as the kernel function (also known as a "smoothing function"), which possesses the six fundamental properties required of a kernel function, including compactness, normalization, and non-negativity. The expression for the cubic spline kernel function is:

$$W_{ij} = \alpha_d \begin{cases} \frac{2}{3} - q^2 + \frac{1}{2} q^3 & 0 \le q < 1 \\ \frac{1}{6}(2-q)^3 & 1 \le q < 2 \\ 0 & q \ge 2 \end{cases} \tag{8}$$

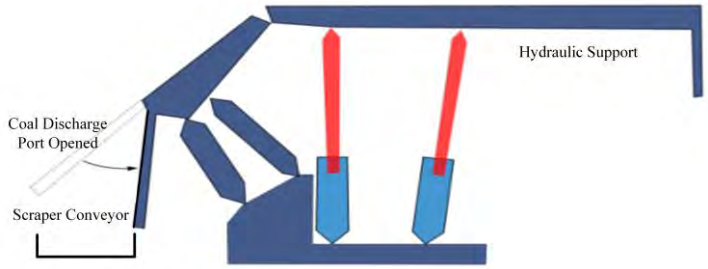
The term q represents a dimensionless distance, given by $q = r/h$, where r is the distance from the origin of the distance function. The regularization coefficient α_d ensures the normalization of the kernel function and in three-dimensional space is expressed as $\alpha_d = 3/2\pi h^2$. In SPH, pairwise interacting particles are typically defined as particle pairs. The kernel function value for a pair of interacting particles i and j can be expressed as $\mathrm{W}(|\boldsymbol{x}_i - \boldsymbol{x}_j|, h_{ij})$ or $\mathrm{W}(|\boldsymbol{x}_{ij}|, h_{ij})$, where $h_{ij} = \frac{h_i + h_j}{2}$.

In this study, the SPH simulation of granular materials adopts an elastoplastic constitutive model that satisfies the Drucker-Prager yield criterion[25-26]. The material deforms elastically until it reaches the yield point, after which it enters the plastic deformation stage.

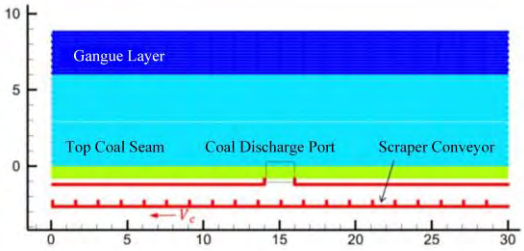
3. Results and Discussion

3.1 Analysis of Coal-rock Interface Evolution Patterns

This section constructs a single coal discharge SPH model, as shown in Figure 2. The model includes a coal layer, a gangue layer, and a scraper conveyor. The scraper conveyor is constructed using virtual particle technology. The scraper is movable, and its speed is determined by the speed of the scraper conveyor. The bottom plate of the scraper conveyor is a single-layer fixed virtual particle. A contact algorithm is used between the virtual particles constituting the scraper base and the scraper and the SPH coal particles and gangue particles. When the coal or gangue particles approach the scraper edge, the virtual particles provide a repulsive force, which exerts a pushing effect on the coal and rock particles[27]. In the single coal discharge model shown in Figure 2, the thickness of the top coal layer is 6.0m, and the thickness of the gangue layer is 3.0m. At the initial moment, the coal discharge port is closed (four layers of virtual particles constitute the coal discharge port); after 20,000 steps of gravity loading, the initial stress distribution of the coal and rock is established, and then the coal discharge port is opened (the four layers of virtual particles within the width range of the coal discharge port are removed). The initial particle spacing in the model is set to 0.2, generating a total of 7668 particles, including 1261 virtual particles. The time step is determined to be 0.0003s based on the CFL criterion. The coal discharge height (distance between the coal discharge port and the scraper conveyor) is set to 1.5m, the spacing between scrapers is 1.5m, and the speed of the scraper conveyor is set to 0~1.5m/s.



(a) Schematic diagram of support and scraper conveyor



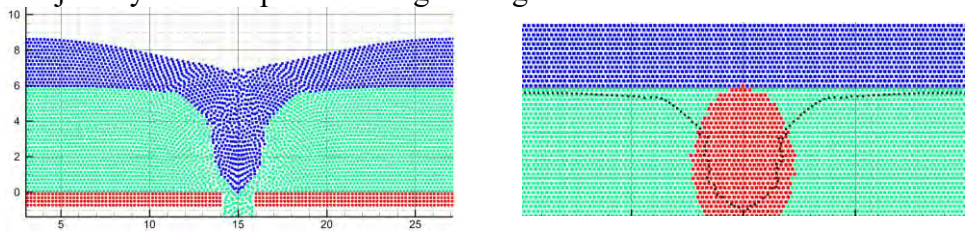
(b) SPH model

Fig 2. Schematic diagram of a single coal discharge SPH model

Along the trend direction of the working face, the hydraulic support structure does not affect the shape of the coal body during the coal mining process. Therefore, as shown in Figure 2b, a numerical model without supports is used to study the top coal caving process. In the direction parallel to the working face, the coal mining width of the hydraulic support is 2.0 meters. When the support is mining coal, the solid boundary virtual particles within the middle 2.0 meters width range of the model's bottom boundary are removed to simulate the

opening of a single coal mining port. The termination condition for coal mining at the current mining port is "seeing gangue and closing the gate," that is, when gangue particles are detected, the mining port is closed by rearranging the 2.0-meter-wide solid boundary virtual particles. That is, when the discharged gangue particles are detected at the coal mining port, the calculation is terminated, and the coal-rock interface at that time is taken as the terminating coal-rock interface.

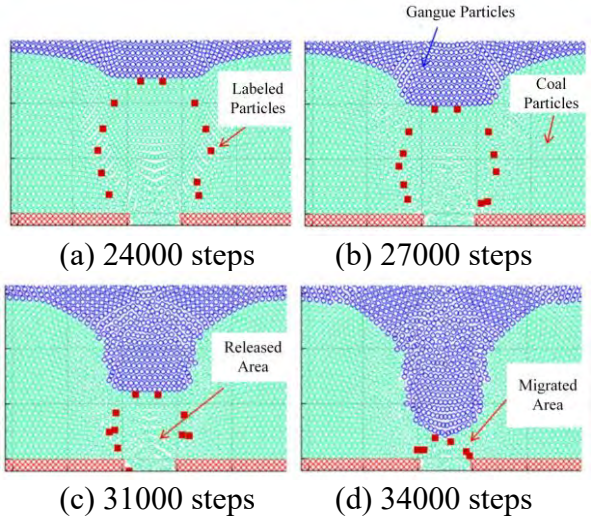
As shown in Figure 3a, after the top coal is extracted, the interface between the coal and rock presents a funnel shape for a single mining panel. To study the shape of the mined-out area and the trajectory of the top coal, a particle labeling method was employed. After the top coal is extracted, the IDs of all released particles are calculated, and then all particles with the same ID are labeled in the numerical model before mining. As shown in Figure 3b, the shape of the mined-out area of the top coal is obtained, which is approximately elliptical in a profile parallel to the coal seam. The particles along the elliptical boundary are labeled to track the movement trajectory of the top coal during mining.



(a) Terminated coal-rock interface (b) Released body

Fig 3. Coal-rock interface and shape of released body at the termination of coal discharge

In different mining steps, as shown in Figure 4, the positions of labeled particles in the crushed top coal are marked, and the overall migration trajectory of all labeled particles throughout the mining process is shown in Figure 5a.



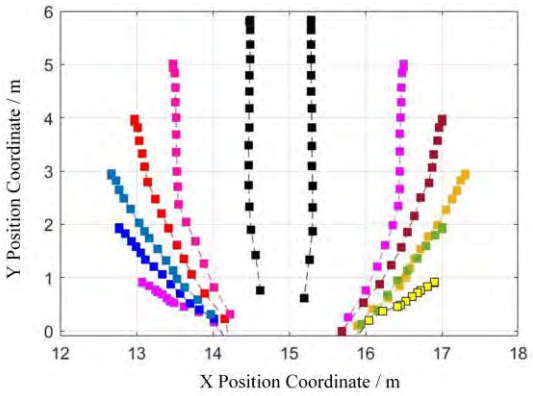
(a) 24000 steps (b) 27000 steps

(c) 31000 steps (d) 34000 steps

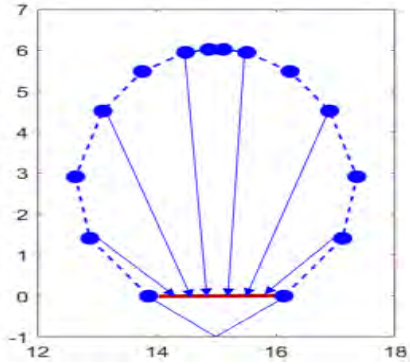
Fig 4. Schematic diagrams of the positions of marked particles at different coal releasing moments.

As shown in Figure 4, at different coal release times, the labeled particles still exhibit an approximate elliptical shape in spatial distribution, and the top coal within this ellipse can eventually be released through the mining panel. Furthermore, the portions of the top coal particles outside the elliptical boundary, although loosened, are not extracted. As shown in Figure 4a, due to the collision and blocking effects of the top coal particles during the mining

process, the labeled particles do not strictly move along a straight line from the elliptical boundary of the top coal to the mining panel. However, the migration trajectory generally conforms to the theoretical model in Figure 5b. Therefore, by comparing Figures 3, 4 and 5, it can be concluded that, in the direction parallel to the working face, the simulation of coal extraction using a single support bracket is generally consistent with the theoretical model.



(a) Top coal movement trajectory

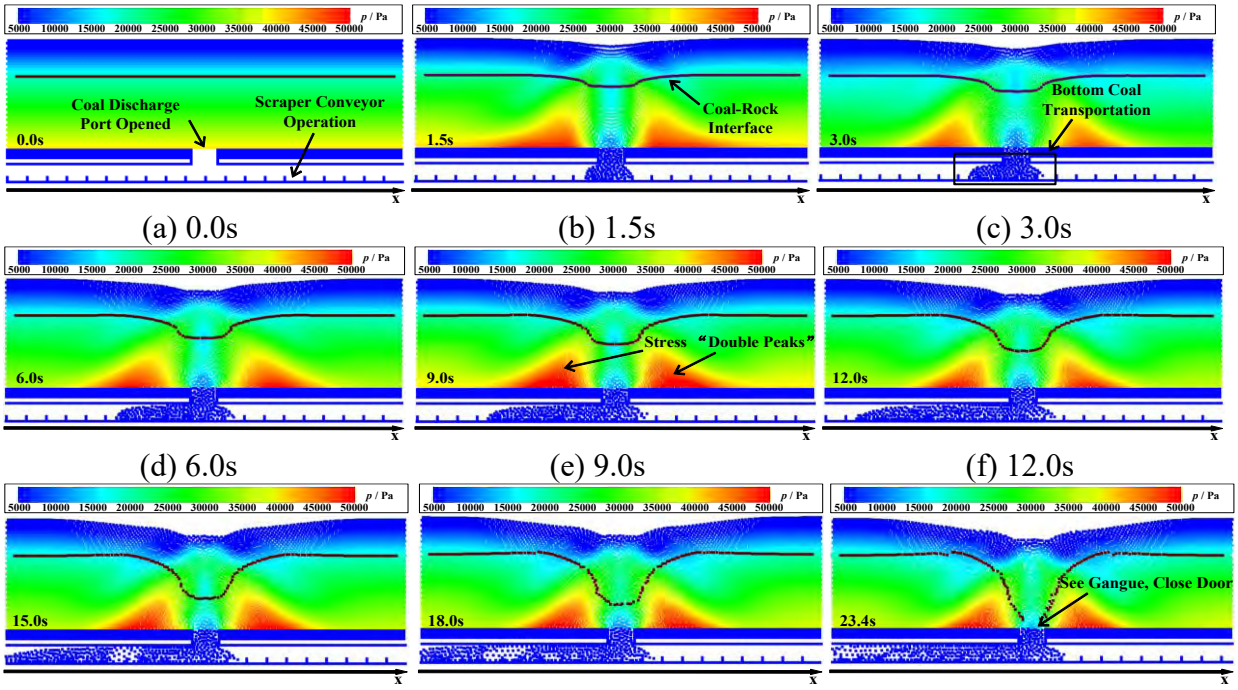


(b) Theoretical model of top coal released body contour

Fig 5. Trajectory of top coal movement and contour of released body

3.2 Analysis of the Impact of Scraper Conveyor Speed on Coal Discharge Process

Figure 6 illustrates the particle distribution and static pressure distribution at different times during the coal discharge process. As shown in the figure, after the coal discharge port is opened, the scraper conveyor starts running in the negative direction of the x-axis at a speed of 1.0 m/s. In Figure (b), the coal near the discharge port falls rapidly under gravity, impacting the surface of the scraper conveyor and spreading to both sides; under the drive of the scraper, the bottom coal moves to the left. At the same time, the coal-rock interface (indicated by the black line in the figure) continues to evolve downward. Coal particles are continuously discharged from the discharge port, and the coal accumulated at the bottom of the port is continuously removed by the scraper conveyor, forming a quasi-steady state of coal discharge. The colors in the figure represent the numerical values of the static water pressure, and it can be seen from Figure (b) that after the end of gravity loading, the static water pressure gradually increases from top to bottom along the numerical direction in a linear distribution; after the start of coal discharge, the pressure is higher on both sides of the discharge port, forming a "double peak" shape, and a basically stable stress distribution is formed in the subsequent coal discharge process. Due to the use of stress smoothing technology in this paper, the static water pressure and stress show a good distribution effect (with small numerical noise). At 23.4 seconds, the discharge port detects gangue particles, and the coal discharge process is terminated.



(a) 0.0s (b) 1.5s (c) 3.0s

(d) 6.0s (e) 9.0s (f) 12.0s

(g) 15.0s (h) 18.0s (i) 23.4s

Fig 6. SPH simulation results of roof coal release and bottom coal conveying process (V_c= 1.0 m/s)

Figure 7 illustrates the relationship between the coal flow velocity at the discharge port and the operating time of the scraper conveyor at different speeds. As shown in the figure, the coal flow velocity rapidly increases to a peak at the start of coal discharge (20000 steps). This is because the opening of the discharge port creates a void below it, allowing the coal particles near the discharge port to free fall, leading to a rapid increase in velocity. However, as the coal particles accumulate below the discharge port, blocking the outlet for coal discharge, the coal flow velocity rapidly decreases and tends to a more stable value. This indicates that as the coal discharge progresses with the support of the scraper conveyor, the coal flow velocity at the discharge port reaches a quasi-steady state. The higher the speed of the scraper conveyor, the greater the stable coal flow velocity. This suggests that adjusting the speed of the scraper conveyor can increase the coal discharge velocity, facilitating the rapid completion of the coal discharge process at the discharge port.

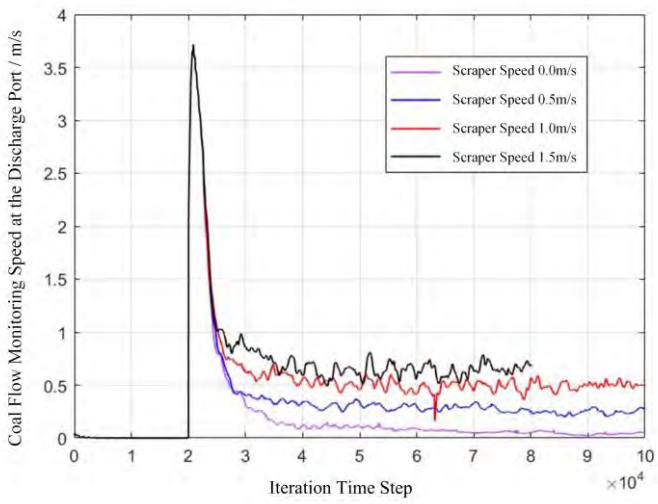
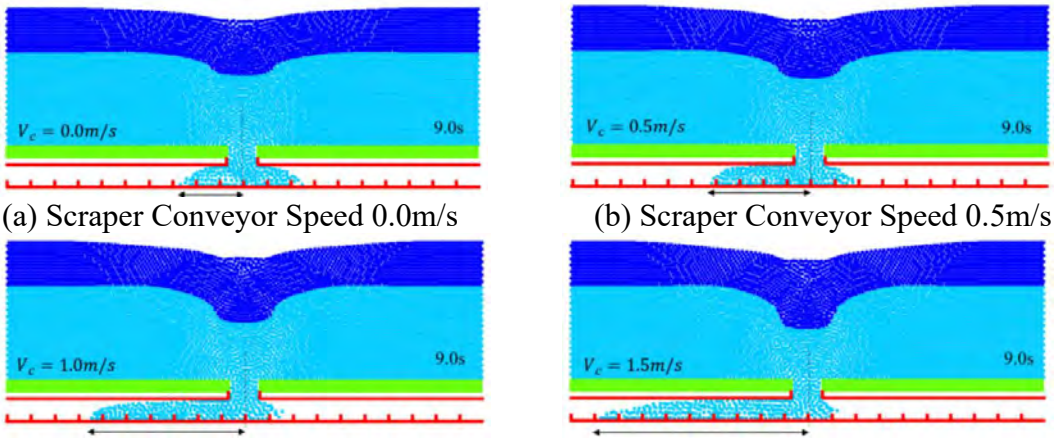


Fig 7. Coal flow velocity-time curve monitored at the coal outlet

This study compared the effects of different scraper conveyor speeds on the coal discharge process. As shown in Figure 8, the simulation results at a certain time (9.0s) for the distribution of top coal, gangue, and discharged coal at different scraper conveyor speeds are compared. In Figure 8a, when the scraper conveyor speed is 0.0 m/s, the scraper is stationary, and the bottom coal transport stops. The coal discharged from the coal outlet accumulates in the gap between the scraper conveyor and the support, hindering the discharge of the remaining coal. The bottom coal is symmetrically distributed above the scraper conveyor relative to the coal outlet. With the increase in the scraper conveyor speed, at the same time of 9.0s, the distance of bottom coal transport to the right increases, the downward evolution distance of the coal-rock interface gradually increases, more coal is discharged, indicating an increase in the coal discharge rate.

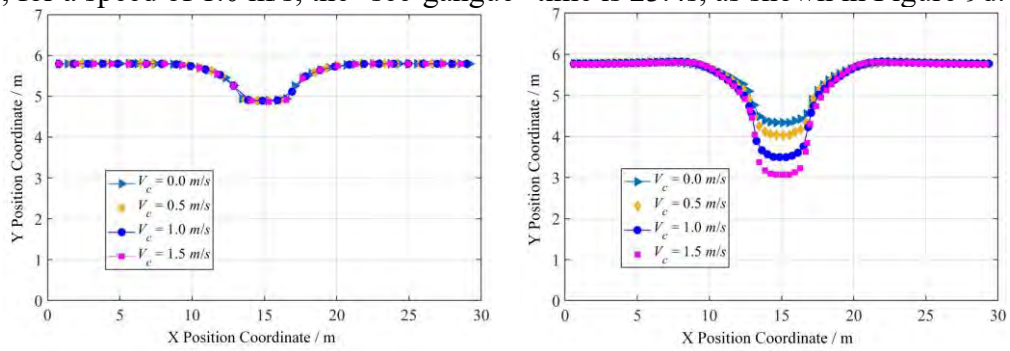


(a) Scraper Conveyor Speed 0.0m/s (b) Scraper Conveyor Speed 0.5m/s

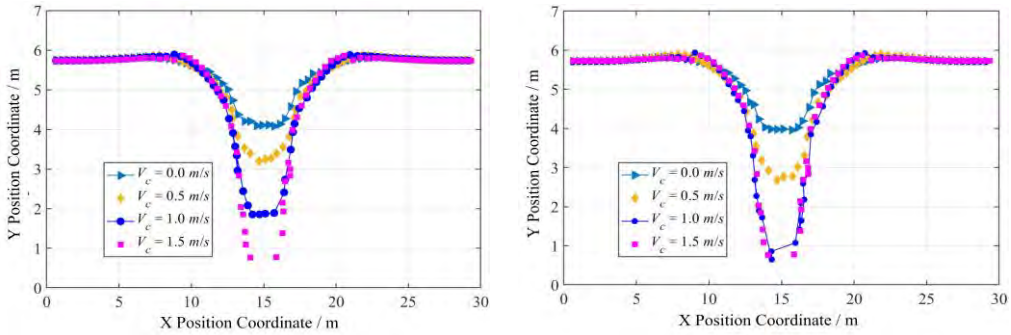
(c) Scraper Conveyor Speed 1.0m/s (d) Scraper Conveyor Speed 1.5m/s

Fig 8. Schematic diagram of the coal-rock interface and the bottom coal conveying distance at different scraper conveyor operating speeds (t=9.0s)

Figure 11 presents the comparative simulation results of the coal-rock interface contour curves at 1.5s, 9.0s, 18.0s, and 23.4s for four different scraper conveyor speeds V_c (0.0 m/s, 0.5 m/s, 1.0 m/s, 1.5 m/s). It can be seen from the figure that with the increase in the scraper conveyor speed, the downward evolution speed of the coal-rock interface also gradually increases. A higher scraper conveyor speed corresponds to a faster coal discharge time (i.e., "see-gangue" time). As shown in Figure 11c, at 18.0s, the coal-rock interface corresponding to a scraper conveyor speed of 1.5 m/s has reached the bottom, indicating "see-gangue." In contrast, for a speed of 1.0 m/s, the "see-gangue" time is 23.4s, as shown in Figure 9d.



(a) t=1.5s (b) t=9.0s

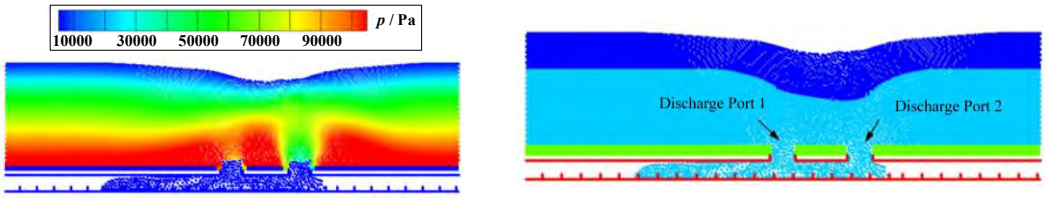


(c) t=18.0s (d) t=23.4s

Fig 9. Schematic diagram of the coal-rock interface and bottom coal conveying distance under different scraper conveyor operating speeds.

3.3 Simulation Results and Comparison of Sequential Coal Discharge along the Working Face Direction

Figure 10 presents the simulation results of simultaneous coal discharge from two coal outlets. The spacing between the two outlets is set to 6.0 meters. The coal particles discharged from outlet 1 will pass through outlet 2 under the action of the scraper conveyor, interfering with the coal discharged from outlet 2. Figure 11 shows the coal flow velocity-time curves for the two outlets. It can be observed that the stable coal flow velocity from outlet 1 is lower than that from outlet 2, indicating that the bottom coal discharged from outlet 1 has a blocking effect on outlet 2. In comparison, the coal flow velocity from outlet 2 is higher than that from outlet 1 but lower than the discharge speed of a single outlet. The pressure distribution results in Figure 10a indicate that due to the higher coal flow velocity from outlet 2, the pressure "double peaks" are more pronounced at outlet 2, leading to different evolutions of the coal-rock interface for outlets 1 and 2, as shown in Figure 10b. This suggests that in practical coal mining processes, simultaneous operation of two outlets requires consideration of interference from the bottom coal.



(a) Pressure distribution (b) Distribution of coal-rock interface

Fig 10. Simulation results of coal being released simultaneously from two coal outlets (t=11.4s)

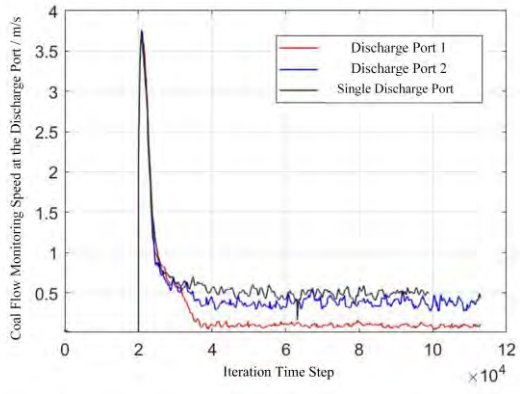
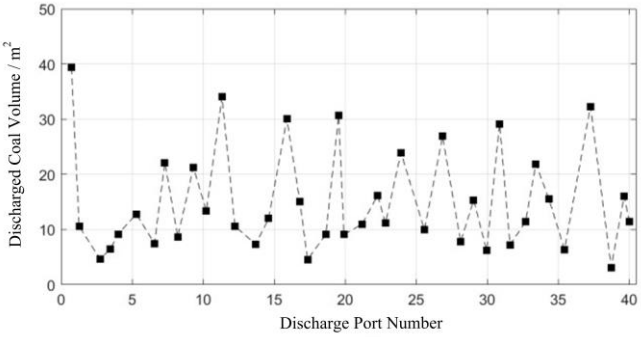


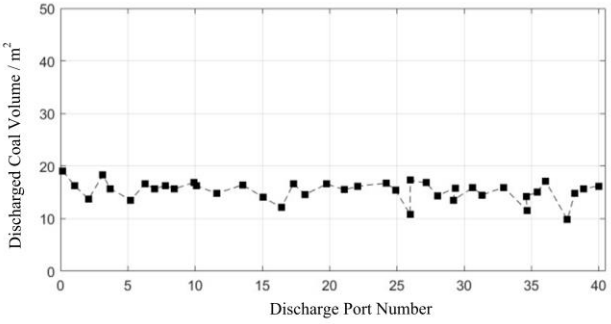
Fig 11. Comparison of coal flow velocity between single and double coal outlets

Further investigations were conducted into the coal discharge process using multiple discharge ports parallel to the working face. The width of the working face was set to 100 meters, and the width of each support discharge port was set to 2.0 meters. A 2D model was used to study the overall top coal discharge effect of 40 support discharge ports within a single coal mining cycle. The discharge process for the 40 ports was simulated from left to right. Traditional top coal caving technology uses the "see gangue, close door" method, ensuring no rock mixes with the discharged top coal. In the automatic coal discharge process, the discharge port is controlled by adjusting the coal discharge time, which allows rock particles to mix with the discharged top coal. Therefore, to further analyze the coal-rock mixture and top coal recovery rate, the discharged coal and rock particles from the 40 ports were collected. The discharge volumes of coal and rock from the 40 discharge ports were statistically compared. The comparison between traditional and automatic coal discharge technologies regarding top coal discharge volume is shown in Figure 12. The recovery rate of top coal for automatic discharge was 85.8%, higher than the recovery rate of traditional discharge (80.9%). However, the rock mixing rate for automatic discharge (7.2%) was also higher than that of traditional discharge (0.0%).

For the automatic coal discharge process, the average top coal discharge volume from the 40 chutes was 17.25m², which was greater than the average top coal discharge volume of traditional coal discharge (16.25m²). Furthermore, the standard deviation of the top coal discharge volume (1.93m²) in the automatic coal discharge process was smaller than that of traditional coal discharge (7.52m²). Although some rock may mix into the discharged top coal during the automatic coal discharge process, the larger average top coal discharge volume in automatic coal discharge leads to a higher top coal recovery rate compared to traditional coal discharge. The smaller standard deviation of top coal discharge volume in automatic coal discharge, along with more uniform discharge from each chute, simplifies operation for workers and improves the efficiency of the scraper conveyor transportation. Overall, automatic coal discharge technology is superior to traditional top coal caving technology.



(a) Traditional coal cutting technique



(b) Automatic coal cutting technology

Fig 12. The amount of roof coal released for each slot using different technologies

4. Conclusions

This study constructs a grid-free numerical model based on the Smoothed Particle Hydrodynamics (SPH) method to successfully simulate the collapse, movement, and extraction processes of top coal in fully mechanized coal mining. The model employs the governing equations of continuum mechanics, integrates the elastoplastic constitutive relation of soil and the Drucker-Prager yield criterion, and effectively realizes the dynamic simulation of bulk top coal. By building a model of the shearer and considering the process of multi-support top coal cutting and bottom coal transportation along the working face, this study reveals the evolution laws of the coal-rock interface and coal extraction rate under different shearer speeds. It is found that the elastoplastic constitutive relation of non-cohesive soil can effectively simulate the particle flow behavior, and the stress of the top coal near the coal cutting face shows a "double peak" distribution characteristic. The practicality and accuracy of the model are verified, as it only requires setting basic material parameters such as friction angle and modulus, thereby avoiding the uncertainty of parameter selection in traditional particle models. Additionally, the study finds that the shearer speed has a significant effect on the coal extraction time, but a relatively small effect on the termination coal-rock interface under the "see gangue close" criterion. Moreover, the "see gangue close" criterion results in differences in the extraction amounts among different cutting faces, with the standard deviation of the top coal extraction volume for 40 cutting faces (7.52m²) significantly higher than that of automatic coal cutting (1.93m²), indicating an influence on the uniformity of coal extraction.

5. References

[1] Yang, S. L., Zhang, J. W., Chen, Y., and Song, Z. Y. (2016) Effect of upward angle on the drawing mechanism in longwall top-coal caving mining, *International Journal of Rock Mechanics and Mining Sciences* **85**, 92–101.

[2] Wang, J. C. (2018) Engineering practice and theoretical progress of top-coal caving mining technology in China, *Journal of China Coal Society* **43**, 43–51.

[3] Wang, G. F., Pang, Y. H., and Ma, Y. (2018) Automated mining technology and equipment for fully-mechanized caving mining with large mining height in extra-thick coal seam, *Coal Engineering* **50**, 1–6.

[4] Wang, G. F., Fan, J. D., Xu, Y. J., and Ren, H. W. (2018) Innovation progress and prospect on key technologies of intelligent coal mining, *Journal of Mine Automation* **44**, 5–12.

[5] Wang, J. C., and Fu, Q. (2002) The loose medium flow field theory and its application on the longwall top-coal caving, *Journal of China Coal Society* **27**, 337–341.

[6] Wang, J. C., Yang, J. L., Liu, H. H., Zhao, D. P., and Zheng, L. Y. (2010) The practical observation research on loose medium flow field theory on the top-coal caving, *Journal of China Coal Society* **35**, 353–356.

[7] Xu, Y. X., Wang, G. F., Li, M. Z., Zhang, J. H., and Han, H. J. (2019) Structure coupling between hydraulic roof support and surrounding rock in extra-thick and hard coal seam with super large cutting height and longwall top coal caving operation, *Journal of China Coal Society* **44**, 1666–1678.

[8] Zhang, N. B., Liu, C. Y., and Chen, Y. M. (2014) Study on coal and gangue flow rule of top-coal caving with erratic hick coal seam by PFC2D, *Coal Technology* **33**, 1–4.

[9] Wang, J. C., Zhang, J. W., Yang, S. L., and Song, Z. Y. (2015) 3-D movement law of top-coal in near horizontal coal seam with multi-gangue under caving mining technique, *Journal of China Coal Society* **40**, 979–987.

[10] Sun, Q., Dan, C. F., Li, Y. F., Wang, J. J., Zhang, H., Zhang, J. Q., and Wu, Z. Y. (2022) Analysis of coal drawing law and parameter research in shallow buried double hard and extra-thick coal seam, *Journal of Mine Automation* **48**, 61–69.

[11] He, X., Liu, C. Y., Wu, F. F., Peng, K. P., and Yang, J. X. (2021) Effect of elevation angle of coal seam on top-coal caving law in fully-mechanized top-coal caving mining face during topple mining, *Coal Science and Technology* **49**, 25–31.

[12] Zhang, J. W., Wang, J. C., and Wei, W. J. (2018) Effect of face dip angle on the drawing mechanism in longwall top-coal caving mining, *Journal of China University of Mining & Technology* **47**, 805–814.

[13] Feng, Y. F. (2020) Research on top-coal caving mechanism and control technology in extra-thick coal seam with hard dirt band, *Coal Science and Technology* **48**, 133–139.

[14] Zou, G. H., Yang, J. N., Guan, S. F., Jiang, Z. G., Wang, M. Q., and Shi, H. Y. (2024) Characterization and simulation of the coal-rock boundary equation on the upper and lower sides of the inclined coal caving opening, *Coal Science and Technology* **23**, 1–13.
[15] Jiang, Z. G., Guan, S. F., Wang, M. Q., Yang, J. N., Shi, H. Y., and Yang, C. Y. (2023) Asymmetric distribution characteristics of coal-rock boundary and optimization of coal caving technology in fully mechanized caving face with large dip angle, *Coal Science and Technology* **42**, 104–108.
[16] Wang, S., Zhang, C. H., He, F., and Yang, Y. (2023) Numerical modelling of loose top coal and roof mass movement for a re-mined seam using the top coal caving method, *PLoS One* **18**, e0283883.
[17] Liu, Y., Wang, J. C., Yang, S. L., Li, L., and Wu, S. (2023) Improving the top coal recovery ratio in longwall top coal caving mining using drawing balance analysis, *International Journal of Mining, Reclamation and Environment* **37**, 319–337.
[18] Wei, W. J., Yang, S. L., Li, M., Zhang, J. W., and Wei, C. B. (2022) Motion mechanisms for top coal and gangue blocks in longwall top coal caving (LTCC) with an extra-thick seam, *Rock Mechanics and Rock Engineering* **55**, 5107–5121.
[19] Zhang, W. H., Li, D. Y., Wang, S., Li, H. M., Zhu, S. T., Sun, K. X., and Zhang, W. Y. (2022) Study on block-particle coupling approach for modeling progressive failure of top coal in extra thick coal seams, *Journal of Henan Polytechnic University* **41**, 24–35.
[20] Zhang, Q. L., Yue, J. C., Liu, C., Feng, C., and Li, H. (2019) Study of automated top-coal caving in extra-thick coal seams using the continuum-discontinuum element method, *International Journal of Rock Mechanics and Mining Sciences* **122**, 104033.
[21] Li, D. Y., Wang, Y. C., Wang, S., Zhu, S. T., and Wang, Z. G. (2023) Numerical simulation study of top-coal loss mechanism and reasonable cycle step length for extra-thick coal seam top-coal drawing, *Journal of Henan Polytechnic University* **42**, 1–10.
[22] Liu, C., Li, H. M., Ma, Z. Y., and Zhang, Z. R. (2023) Numerical simulation study on reasonable top coal caving technology in longwall top coal caving working face with different top coal thickness, *Journal of Henan Polytechnic University* **42**, 40–46.
[23] Dong, X. W. (2021) *Extension of Meshless Weakly Compressible SPH Numerical Algorithms and Applications*, Science Publishing House, Beijing.
[24] Zhang, W., Yan, F., Wang, Z. F., and Li, S. J. (2024) Numerical analysis oflandslides based on coupling modelof material point method and depth integral, *Rock and Soil Mechanics* **45**, 1–11.
[25] Feng, R. F., Fourtakas, G., Rogers, B. D., and Lombardi, D. (2021) Large deformation analysis of granular materials with stabilized and noise-free stress treatment in smoothed particle hydrodynamics (SPH), *Computers and Geotechnics* **138**, 104356.
[26] Hu, M., Xie, M. W., and Wang, L. W. (2016) SPH simulations of post-failure flow of landslides using elastic-plastic soil constitutive model, *Chinese Journal of Geotechnical Engineering* **38**, 58–67.
[27] Gao, T., Xie, S. Y., Hu, M., Tan, Q. T., Fang, H. Z., Yi, C., and Bao, A. H. (2022) Soil-soil engaging component SPH model based on a hypoplastic constitutive model, *Transactions of the Chinese Society of Agricultural Engineering* **38**, 47–55.

SPH numerical simulation of interaction between different types of baffles and liquid sloshing

***†Heng Zhang[1], Bo Liu[1], Hanxiang Wang[1], Yanxin Liu[1], Xiangwei Dong[2], and Yu Zhang[1]**

[1]China University of Petroleum (East China)

[2]Shandong University of Science and Technology

*Presenting author:jhdxpz@163.com

†Corresponding author: jhdxpz@163.com

Abstract:

To suppress the liquid sloshing phenomenon in the tanks of liquid-carrying ships and optimize their structure, this study employs the continuity equation and Navier-Stokes (N-S) equation for viscous fluids as the governing equations. These equations are discretized using the Smoothed Particle Hydrodynamics (SPH) method. An algorithm is developed to numerically simulate the liquid sloshing in a two-dimensional rectangular tank at a 20% liquid loading rate. The forces generated by sloshing in tanks with different types of baffles are calculated and compared for the same loading rate.

Keywords: Liquid sloshing, Fluid-solid coupling, Baffles, Numerical simulation

Introduction

Continuous innovation in marine engineering technology and the sustained development of marine vessels have driven exploration and exploitation activities of marine resources into deeper sea areas. In these complex marine environments, ships such as large oil tankers and liquefied natural gas (LNG) carriers may experience severe liquid sloshing under harsh sea conditions[1]. This sloshing can cause localized thumping damage to bulkheads and potentially lead to liquid cargo leakage, resulting in significant economic losses and severe environmental pollution. To mitigate such damage, one effective solution is to install baffles within the liquid tanks. Studying the interaction between different baffle configurations and liquid sloshing is crucial for addressing the engineering risks associated with liquid sloshing[2].

1. Mathematical model

1.1. Governing equation

（1） Continuity equation

The multiphase flow control equation used in this study is based on the Smoothed Particle Hydrodynamics (SPH) model proposed by Zhang[3] et al. , which is suitable for gas-liquid multiphase flow with a high density ratio. To address the sharp changes in particle mass and density across the high-density interface, this paper employs a particle approximation equation based on particle volume. The specific expression is

$$\rho_i = m_i \sum_j W_{ij} \tag{1}$$

ρ_i represents the density of particle i, m_i represents the mass of particle i; W_{ij} is a smooth function value.

（2） Equation of motion

To discretize the pressure gradient term of the momentum equation, considering the sudden change in density across the interface, particle volume is used in place of density. Consequently, the discrete form of the pressure gradient term in the equation can be reformulated as follows:

$$(\nabla P)_i = -\frac{1}{V_i} \sum_j (P_i V_i^2 + P_j V_j^2) \nabla_i W_{ij} \tag{2}$$

In the formula, P_i and P_j represent the pressure of particle i and particle j respectively, and the unit is Pa, V_i and V_j represent the volumes of particles i and j in m^2.

1.2. Equation of state

Since the Navier-Stokes (N-S) equation is not closed, it is necessary to introduce the pressure equation of state to close the governing equations. In this study, different equation of state constants are used for the light and heavy phases. Here, the subscript l represents the heavy phase, and the subscript g represents the light phase. The equations of state for the heavy and light phases are expressed as follows:

$$P_l = \frac{c_{0l}^2 \rho_{0l}}{\gamma_l} \left(\left(\frac{\rho}{\rho_{0l}} \right)^{\gamma_l} - 1 \right) + P_b \tag{3}$$

$$P_g = \frac{c_{0g}^2 \rho_{0g}}{\gamma_g} \left(\left(\frac{\rho}{\rho_{0g}} \right)^{\gamma_g} - 1 \right) + P_b \tag{4}$$

Where, the coefficients of the equation of state for the heavy and light phases are γ_i=7 and γ_g =1.4, respectively. ρ_{0l} and ρ_{0g} reference densities of heavy and light phases, in kg/m^3, respectively; c_{0l} and c_{0g} are reference values for the artificial sound velocity of the heavy and light phases, respectively, in m/s.

2. Numerical simulation of baffle-liquid slosh interaction

2.1. Calculation model specification

In this study, the rectangular tank model is used as the research object. To ensure the research has significant engineering relevance, the design of the central cargo tank of a 9000DWT tanker is referenced[4], and appropriate simplifications are made to build a model suitable for computational analysis. This paper primarily analyzes the effect of rolling excitation on liquid sloshing. For efficient numerical calculation, the model dimensions are set as length L = 0.5m, width W=0.15m, and height H=0.3m[5]. The model is shown in Figure 1:

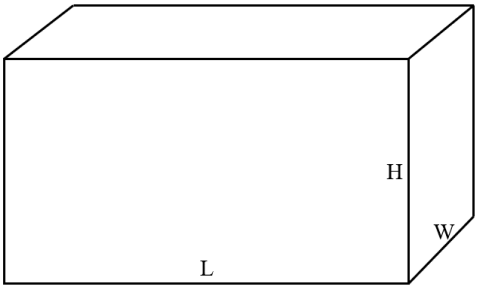


Figure 1. Schematic diagram of tank model

To ensure the transportation safety of the tank, and in accordance with the general liquid-carrying requirements specified for liquid-carrying ships, a liquid carrying rate of 20% was selected for numerical simulation[7].

The natural frequency of the tank is calculated using the following empirical formula[6]:

$$\omega_{i,r} = \sqrt{g\frac{\pi i}{B_T}\tan h(\frac{\pi i}{B_T}h)} \tag{5}$$

i is the modal order, B_T is the tank length, and h is the liquid carrying depth.

When the external excitation frequency equals the natural frequency of the tank, the swaying is most intense. Therefore, to study the swaying phenomenon of the ship under potentially dangerous conditions, this paper sets the liquid loading conditions such that $\omega=\omega_0$[8][9], the external excitation frequency is equal to the natural frequency.

2.2. Influence of different height baffles on liquid sloshing

（1） Description of calculation model

An anti-sway baffle is installed at the center of the tank's bottom, with a width of W= 0.02m. The tank model is shown in Figure 2. This section primarily examines the effect of baffle height on liquid sloshing. The baffle heights considered are 0.03m, 0.05m, 0.06m and 0.07m respectively. Two pressure measuring points are installed on the right bulkhead and the right bilge of the tank model. The setup and distribution of these pressure measuring points are detailed in Table 1.

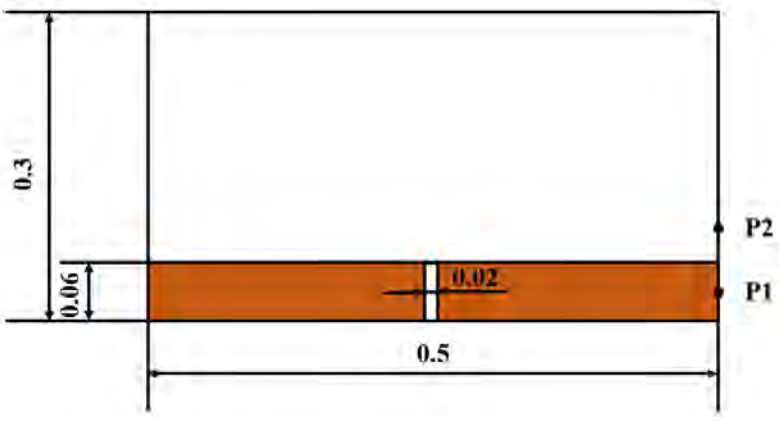


Figure 2. Rectangular tank models with different flexible baffles (unit: m)

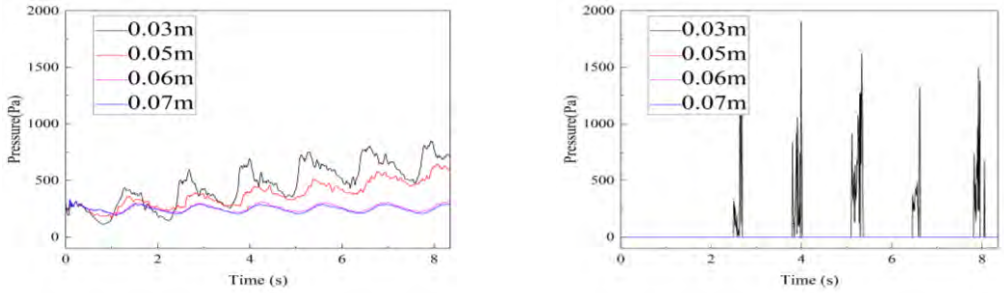
Table 1. Position table of pressure measuring points

pressure tap	x coordinate /m	y coordinate /m	z coordinate /m
P1	0.25	0	0.03
P2	0.25	0	0.09

The tank is rolling, Its equation of motion is $\theta(t) = \theta_{\max} \sin(\omega t)$,Excitation frequency is f=0.776 Hz,Rotation amplitude is $\theta_{\max}$ =3°.

（2） Calculation results and analysis

In Figure 3, the pressure change curves for each pressure measuring point of the tank equipped with baffles of different heights are depicted under the condition of a 20% liquid load rate.

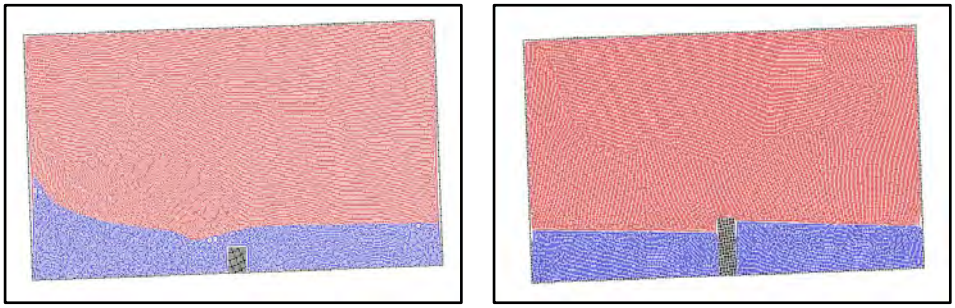


（a）P1 （b）P2

Figure 3. Pressure time history curve of different height baffles at each pressure measuring point

The graph illustrates that the impact pressure on the bulkhead gradually decreases with increasing height of the tank's center baffle. When the baffle height reaches or exceeds the liquid level (e.g., 0.06 m and 0.07 m), the impact pressure during tank swinging significantly diminishes, dropping well below the impact pressure observed when the baffle is lower than the liquid level. This observation highlights the effective inhibition of tank fluid sloshing achieved by increasing the baffle height.

Figure 4 displays the change in the free surface of the liquid with anti-sway damper heights of 0.03 m and 0.07 m simultaneously. As the baffle height increases, the ability to restrain tank sloshing becomes more pronounced. The fluctuation of the liquid surface diminishes, and the liquid surface exhibits a smooth, straight-line characteristic.



（a）h=0.03 m （b）h=0.07 m

Figure 4. Change of free liquid surface at the same time with different baffle height

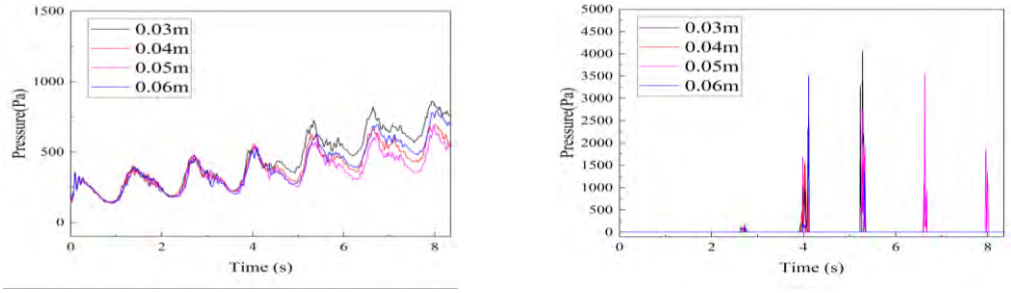
2.3. Influence of different width baffles on liquid sloshing

（1） Description of calculation model

In this section, the same tank model as in the previous section is used, with a baffle height H=0.04m. The focus is on studying the influence of baffle width on liquid sloshing. The baffle widths considered are 0.03 m, 0.04 m, 0.05 m, and 0.06 m. The external motion applied follows the equation of motion $\theta(t)=\theta_{\max}\sin(\omega t)$, with an excitation frequency f=0.776 Hz, and a rotation amplitude $\theta_{\max}$ =3°.

（2） Calculation results and analysis

In Figure 5, the pressure change curves for each pressure measuring point in a tank equipped with baffle plates of different widths are depicted at a liquid load rate of 20%.

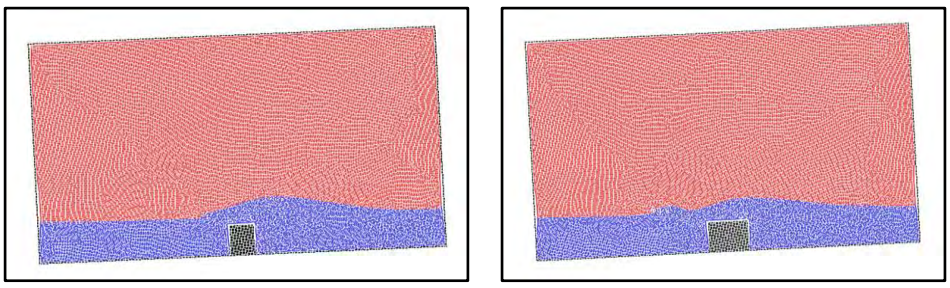


（a）P1 （b）P2

Figure 5. Pressure time history curve of different width baffles at each pressure measuring point

The graph illustrates that as the width of the anti-sway baffle inside the tank increases, there is a gradual decrease in pressure on the right wall of the tank. However, the pressure reduction due to width changes is not significant, even though the barrier effect on fluid flow becomes more pronounced after widening the baffle. This indicates that, at a 20% liquid load rate, simply increasing the baffle width may not be the most effective approach to improving the anti-sway effect and enhancing economic benefits.

Figure 6 displays the change in the free surface of the liquid with anti-sway damper widths of 0.03 m and 0.05 m simultaneously. As the baffle width increases, the sloshing amplitude of the liquid decreases, and the deformation of the free liquid surface becomes smoother. The wavefront forms smaller peaks and troughs, dynamically and periodically changing with the periodic movement of liquid sloshing.



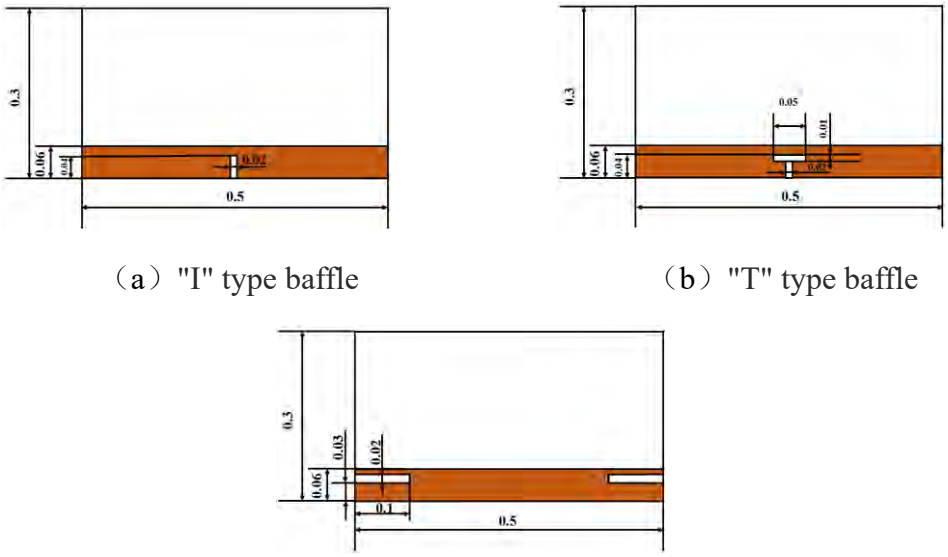
（a）w=0.03 m （b）w=0.05 m

Figure 6. Change of free liquid surface at the same time with different baffle width

2.4. Influence of different shapes of baffles on liquid sloshing

（1） Description of calculation model

In this section, the same tank model as in the previous sections is used, and the tank is equipped with three types of baffle plates: "I" baffle plate, "T" baffle plate, and horizontal symmetrical baffle plate. The specific parameters of these baffles are shown in Figure 7. The external excitation motion used follows the equation $\theta(t)=\theta_{\max}\sin(\omega t)$, with an excitation frequency f=0.776 Hz and a rotation amplitude $\theta_{\max}$ =3°.



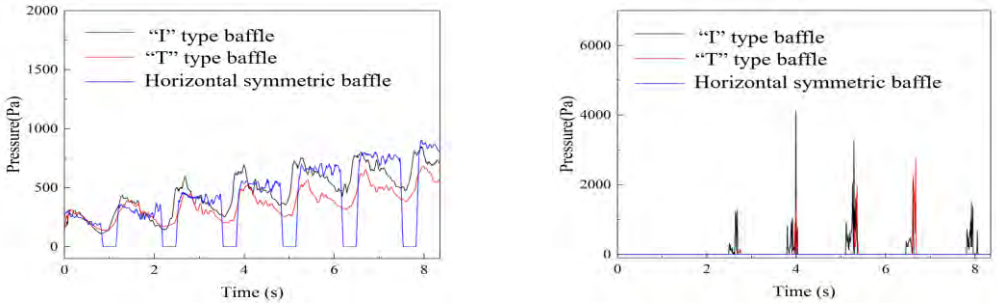
（a）"I" type baffle （b）"T" type baffle

（c）Horizontal symmetric baffle

Figure 7. Rectangular tank model with different shapes of baffles (unit: m)

（2） Calculation results and analysis

In Figure 8, the pressure change curves for each pressure measuring point in the tank equipped with different shapes of baffles are depicted under the condition of a 20% liquid load rate.



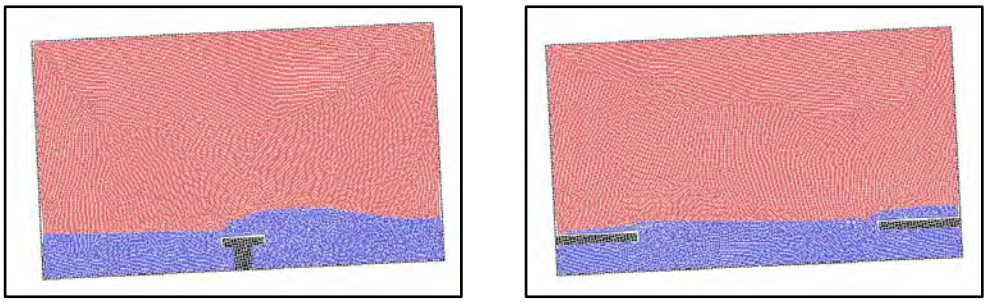
（a）P1 （b）P2

Figure 8. Pressure time curve of each pressure measuring point of different shapes baffles

The curve illustrates that the "T" baffle configuration is more effective in reducing slosh pressure in the tank compared to the "I" baffle configuration. The horizontal symmetrical baffle, when compared to the other two types of baffles, generally reduces sloshing pressure in the

liquid tank. The pressure curve trend at pressure measurement point P1 is different for the horizontal symmetrical baffle, which is influenced by the baffle's position and its obstruction of the upward movement trend of the liquid extending bulkhead, thereby slowing down the fluctuation of the free liquid surface.

Figure 9 demonstrates the changes in liquid level with a "T" shaped baffle and horizontal symmetrical baffle at the same time. It is evident that the horizontal symmetrical baffle significantly inhibits liquid sloshing. The obstruction effect of the horizontal baffle greatly reduces the amplitude of liquid sloshing, resulting in a smooth deformation of the free liquid surface without distinct peaks and troughs.



（a）"T" type baffle　　（b）Horizontal symmetric baffle

Figure 9. Changes of free liquid surface at the same time with different shapes of baffles

3. Conclusion

This paper examines the influence of baffles of different heights, widths, and shapes on liquid sloshing in a tank under a 20% liquid load rate. First, we analyze the effect of varying baffle heights on sloshing. It is observed that at a 20% liquid load rate, the height of the baffle is positively correlated with its ability to dampen oscillations. Higher baffles exhibit a more pronounced dampening effect on liquid sloshing. However, when the baffle height exceeds the free liquid level, the enhancement in sloshing inhibition diminishes with increasing baffle height. This indicates that beyond a certain height, the inhibitory effect on liquid sloshing stabilizes.

Secondly, we investigate the impact of different baffle widths on liquid sloshing. It is found that increasing the baffle width at a 20% liquid load rate moderately reduces tank oscillations. However, this improvement is somewhat limited and may lead to material wastage. Compared to increasing baffle height, widening the baffle is less effective in suppressing sloshing.

Lastly, the study analyzes the influence of different baffle shapes on liquid sloshing. It is discovered that "T"-shaped baffles and "I"-shaped baffles demonstrate superior oscillation damping effects, effectively reducing pressure fluctuations. Horizontal symmetrical baffles show no significant effect on mitigating pressure fluctuations caused by liquid sloshing against the bulkhead, but they do reduce wave topping and maintain a smooth free surface. Therefore, in practical applications, a combination of different baffle shapes can be considered based on specific requirements.

References:

[1] Wen Su. Globalized Oil and Gas Competition and China's Strategic Choices [D]. China University of Geosciences (Beijing), 2012.

[2] Z.H Cai. Study on Swaying Problem of Liquid Tank of Liquid Cargo Ship [D]. Shanghai: Shanghai Jiao Tong University,2012.

[3] Zhang A, Sun P, Ming F. An SPH modeling of bubble rising and coalescing in three dimensions[J]. Computer Methods in Applied Mechanics and Engineering, 2015, 294: 189-209.

[4] J.P. Wang, Z.C. Xiao. 9000DWT Tanker Design [J]. Jiangsu Ship, 2010 (3): 8-10.

[5] R.Q. Zhu, Y.S. Wu. Numerical study of fluid shaking characteristics in liquid tanks [J]. Shipbuilding of China, 2002, 43 (2) :15-28.

[6] Nasar T, Sannasiraj S A, Sundar V. Experimental study of liquid sloshing dynamics in a barge carrying tank[J]. Fluid Dynamics Research, 2008, 40(6): 427.

[7] American Bureau of Shipping. Strength Assessment of Membrane-Type LNG Containment Systems Under Sloshing Loads[M]. 2002.

[8] Y.G. Chen et.al. Numerical simulation of liquid sloshing in LNG tanks using a compressible two-fluid flow model [J]. ISOPE.2009:221-230

[9] Z.Y Fang, S.L. Yang, R.Q. Zhu. Application of the Level-set method to liquid shaking studies [J]. Hydrodynamics Research and Progress Series A, 2007, (02):150-156.

Topology optimization for underwater pressure hull considering buoyancy-weight ratios minimization

***Yuanteng Jiang[1], Tengwu He[1], and †Min Zhao[1,2]**

[1]State Key Laboratory of Ocean Engineering, School of Ocean and Civil Engineering, Shanghai Jiao Tong University, Shanghai 200240, China
[2]Collaborative Innovation Center for Advanced Ship and Deep-Sea Exploration (CISSE), Shanghai 200240, China

*Presenting author: joel-embid@sjtu.edu.cm
†Corresponding author: min.zhao@sjtu.edu.cn

Abstract

As the key component of submersible, the design of underwater pressure hull is particularly important. In this work, based on a parameterized level-set method, a topology optimization method is proposed that can account for the design of underwater pressure hull for submersible. The optimal design of underwater pressure hull is studied by using the buoyancy-weight ratios minimization as the objective function, the critical buckling factor and the yield strength as the design constraint. To provide clear and continuous mathematical expressions for the normal velocities, the theory of the shape derivative and bifurcation analysis are used to obtain the normal velocities in the parameterized level-set method, and we propose an easily implemented method to discretize normal velocities to every nodal point in the design area. An augmented Lagrange multiplier is given to realize stable transitions of both optimization problems during the convergence process. The effectiveness of the proposed method is validated by numerical examples, and it has engineering application value for the exploration of new underwater pressure hulls.

Keywords: Topology optimization, Parameterized level-set method, Buckling optimization, Buoyancy-weight ratios minimization, Underwater pressure hull

Introduction

In the optimization design of deep-sea underwater pressure hull, the buoyancy-weight ratios of underwater pressure hull will greatly affect the overall performance of the submersible. On the other hand, there are usually two forms of failure for pressure hull: material failure and structural instability. Among them, structural instability is also known as structural buckling, which refers to the phenomenon where the stable equilibrium state of the structure begins to lose when the external force increases to a certain amount, and with slight disturbance, the structural deformation will rapidly increase and lead to structural failure. In the previous works to improve the stability of pressure hull, the common methods were to increase the wall thickness, add stiffening ribs, and even set up corrugated walls [1, 2], and the optimization methods adopted usually involved size and simple shape optimization. However, the performance of the pressure hulls obtained by above methods largely depends on the given shape. Therefore, to select the optimal topological form of a pressure hull which satisfy the design requirements need to use a new design method to conduct an in-depth exploration.

For the research of the topology optimization considering stability problems, there are several works carried out. Lindgaard and Dahl [3] proposed a topology optimization method with

buckling analysis based on a non-linear equilibrium equation, which could provide a more accurate description of the buckling phenomenon, at an increased computational cost during optimization. However, it is difficult to determine the sensitivity based on non-linear equilibrium equations, and optimization process has a high computational cost. Thus, linear buckling analysis has been widely employed in topology optimization due to the ease of sensitivity analysis expression and acceptable computational cost. Ferrari et al. [4] proposed 250-line MATLAB open-source code with an improved optimization criterion method considering a buckling constraint. Zhang et al. [5] adopted a feature-driven optimization method and the finite cell method to solve buckling-constraint topology optimization problems. Xu et al. [6] used the BESO method in a proposed compliance-minimization problem under buckling and volume fraction constraints. Xu et al. [7] proposed a novel algorithm for maximizing buckling strength, and stress relaxation functions are employed to eliminate stress singularities in this work. Although the density method is relatively mature in terms of algorithm in buckling-based problems, the low-density areas account for a large proportion during the optimization process [4]. Such interpolation models were highly nonlinear, and difficult to be intuitionally solved when the geometry model is complex. Besides, the initial structure in the optimization process is ambiguous, and the final structure is not unique, depending on the selected interpolation models. In recent years, the level-set method was paid growing attention to the problem of pseudo-buckling modes, resulting from remarkable advantages [8]. Compared to the density method, the structures with the level-set method have clear contours, and there is no large proportion low-density area in the calculation process. Most importantly, level-set method can effectively alleviate the problem of pseudo-buckling modes during the optimization process. However, there are still few studies for buckling-based problems by level-set method, in addition to Deng [8], where a computational scheme for analysis of the velocity field in the level set method by discrete adjoint method is presented while the details of the optimal results were not enough and far less than the previous studies.

In this study, for buoyancy-weight ratios minimization problem considering the critical buckling factor and the yield strength as the design constraints, a novel method on basis of modified parameterized level-set method is proposed, in which the velocity field is continuously described based on the theory of the shape derivative and bifurcation analysis. To ensure numerical stability in the optimization process, a regularization term is added in the parameterized level set method. Furthermore, augmented Lagrange multipliers are developed to make the transitions of optimization problem stable during convergence. Finally, underwater cylindrical pressure hull problem is solved. All in all, using current developed method, the optimal results with clearer contour and more details can be obtained, which is helpful for practical manufacturing

Parameterized level-set method

In the optimization design of deep-sea underwater pressure hull, the buoyancy-weight ratios of underwater pressure hull will greatly affect the overall performance of the submersible. On the other hand, there are usually two forms of failure for pressure hull: material failure and structural instability. Among them, structural instability is also known as structural buckling, which refers to the phenomenon where the stable equilibrium state of the structure begins to lose when the external force increases to a certain amount, and with slight disturbance, the structural deformation will rapidly increase and lead to structural failure. In the previous works to improve the stability of pressure hull, the common methods were to increase the wall thickness, add stiffening ribs, and even set up corrugated walls [1, 2], and the optimization methods adopted usually involved size and simple shape optimization. However, the

performance of the pressure hulls obtained by above methods largely depends on the given shape. Therefore, to select the optimal topological form of a pressure hull which satisfy the design requirements need to use a new design method to conduct an in-depth exploration.

The idea is as follows and is shown in Fig. 1. The zero contour of a high-dimensional scalar function $\phi(x)$, and Hamilton–Jacobi equation let $\phi(x)$ evolve,

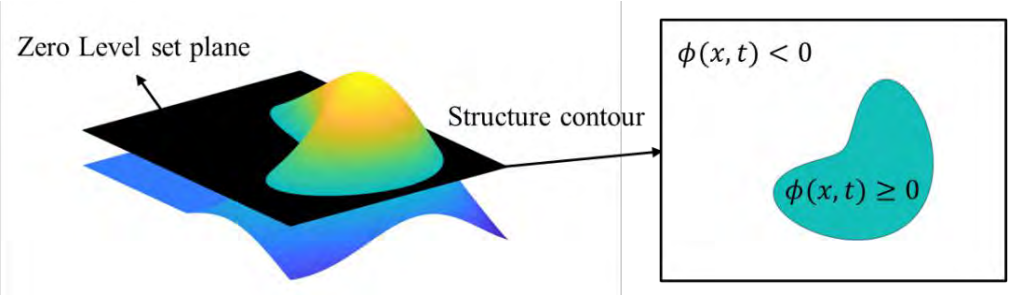
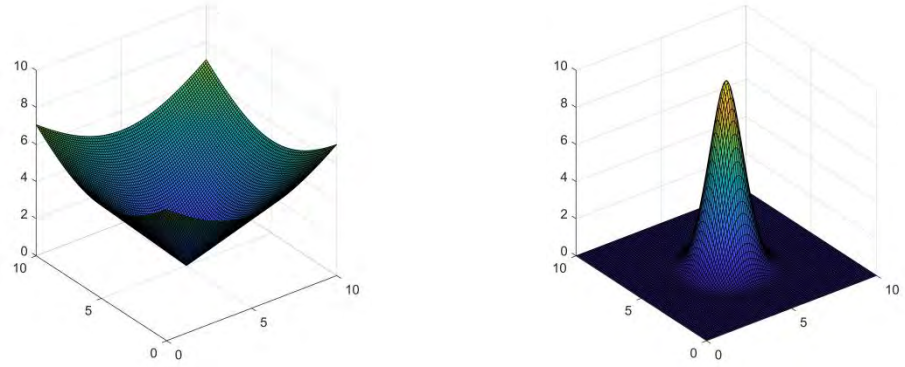


Figure 1: Description of two-dimensional structure by level-set method.

$$\frac{\partial \boldsymbol{\phi}}{\partial t} + \boldsymbol{v_n}|\boldsymbol{\nabla \phi}| = \mathbf{0}, \boldsymbol{\phi}(\boldsymbol{x},\mathbf{0}) = \boldsymbol{\phi_0}(\boldsymbol{x}), \tag{1}$$

where $v_n = \boldsymbol{v}\frac{\boldsymbol{\nabla \phi}}{|\boldsymbol{\nabla \phi}|}$, and v_n is its velocity field, which will determine the level-set function and the structure contour changes with time t. The level-set method is used to solve the Hamilton–Jacobi partial differential equation to complete the topology optimization. However, most of the above level-set approaches use the conventional level-set method, whose topology optimization has poor computational inefficiency [9], prompting several researchers to present parameterized level-set methods based on radial basis functions [10, 11].

The use of a radial basis function to approximate the fitting of sample data has a strong ability to estimate local data, so it is often used to fit or reconstruct the geometric surface of large-scale data. In addition, it will effectively guarantee the smoothness of the approximate surface. The accuracy of radial basis function interpolation and the calculational efficiency in the interpolation process are related to the type of radial basis function selected. There are both globally supported and compactly supported radial basis functions, as shown in Fig. 3, different methods of discrimination are based on the sparsity of the matrix formed during interpolation.



(a) **(b)**

Figure 2: Kinds of radial basis functions: (a) Globally supported; (b) Compactly supported.

With deeper understanding of the characteristics and advantages of radial basis functions, their scope of application has been extended to topology optimization, and the efficiency of the method has prompted researchers to consider the parameterized level-set method based on

radial basis functions. For a set of radial basis functions with sample points, the level-set function interpolated by radial basis functions is

$$\boldsymbol{\phi(x,t) = \varphi(x)\alpha(t)}, \tag{2}$$

where $\phi(x,t)$ is the level-set function at time t, and $\varphi_i(x)$ is the ith radial basis function, with weight coefficients $\alpha_i(t)$ during interpolation. Equation (2) shows that $\phi(x,t)$ has two parts due to the interpolation of the radial basis function. One is a set of radial basis functions $\varphi(x)$ that are only related to the space variable x. The other is a set of weight coefficients $\alpha(t)$ that are only related to the time variable t. In this method, $\phi(x,t)$ completes the decoupling of the time variable t and space variable x. The positions of the sample points of radial basis function group $\varphi(x)$ will not change during optimization, and only the weight coefficient $\alpha(t)$ related to the time variable t changes over time. The expression of parameterized level-set method is

$$\boldsymbol{\varphi(x)\frac{d\alpha(t)}{dt} + v_n|\nabla\varphi(x)\alpha(t)| + \tau\nabla^2\phi = 0, \phi(x,0) = \phi_0(x).} \tag{3}$$

The regularization term $\nabla^2\phi$ is presented in level set method based on reaction diffusion equation [12], and this term ensures the smoothness of the structure contour. The appropriate value of the regularization parameter τ can prevent the generation of excessive details in optimal results, and ensure numerical stability during the optimization process.

Problem formulation

In this section, we construct a topology optimization model with the buoyancy-weight ratios minimization of underwater pressure hull as the objective, and critical buckling factor and the yield strength as the design constraints. The standard mathematical model is

$$\begin{aligned} &\underset{\Omega}{\boldsymbol{minimize}}: && \boldsymbol{R(\Omega) = \frac{\rho_0 \int_\Omega dx}{\rho_w \int_{\Omega_1} dx}} \\ &\boldsymbol{subject\ to}: && \boldsymbol{\int_\Omega Ae(u)e(v)dx = \int_{\Gamma_N} pvds}, \boldsymbol{u,v \in U_{ad}} \\ & && \boldsymbol{\lambda \le 3\bar{\lambda}} \\ & && \boldsymbol{\sigma_{max} \le 0.85\sigma_y} \end{aligned} \tag{4}$$

where Ω is the design area, and Ω_1 is the outside contour of the underwater pressure hull, and $R(\Omega)$ is the buoyancy-weight ratios, and $\boldsymbol{u}$ is the displacement field of the structure, and Γ_N is a Neumann boundary, and $\boldsymbol{p}$ is the pressure of Neumann boundary, and λ is the critical buckling factor, and $\bar{\lambda}$ is the limit of the buckling factor, and $\boldsymbol{\sigma_{max}}$ is the maximal value of von-Mises stress in the structure, and $\boldsymbol{\sigma_y}$ is the yielding strength.

Normal velocities

In this problem, the appearance of the pressure hull remains unchanged, and the maximal value of von-Mises stress in structure always occurs at the position with the minimal wall thickness, and the stress-constraints can be achieved by controlling the minimal value of the wall thickness in the pressure hull. First, to build the bifurcation analysis model,

$$\boldsymbol{\int_\Omega Ae(\psi)e(\psi)dx + \lambda \int_\Omega Ae(u):q(\psi,\psi)dx = 0,} \tag{5}$$

where $\boldsymbol{q}(\psi,\psi) = \psi_{k,i}\psi_{k,j}$, and ψ is the buckling mode of the structure, and λ is the buckling load factor. In the stability problem, the designer hopes to get a structure with a larger λ. To facilitate the modeling of the optimization problem, we take the reciprocal of λ, and Equation (5) can be written as

$$\boldsymbol{\int_\Omega Ae(u):q(\psi,\psi)dx + \mu \int_\Omega Ae(\psi)e(\psi)dx = 0.} \tag{6}$$

where $\mu = \frac{1}{\lambda}$. The maximum value of μ is approximated by the Kresselmeier–Steinhauser (K–S) aggregation function to make sure the differentiability of the expression with parameter ρ_{KS},

$$\boldsymbol{J^{KS}(\mu_i(\Omega)) = \mu_1 + \frac{1}{\rho_{KS}} \ln \left(\sum_{i=1}^{n} e^{\rho_{KS}(\mu_1 - \mu_i)} \right)}, \tag{7}$$

whose purpose is to give an upper bound on μ_1, which is the reciprocal of the fundamental buckling load factor λ_1. The Lagrange function for each reciprocal of the buckling load factor terms can be constructed as

$$\boldsymbol{l(\Omega,u,v) = \mu_i(\Omega) + \int_\Omega Ae(u)e(v)dx - \int_{\Gamma_N} p_0 v ds}, \boldsymbol{u,v \in U_{ad}}, \tag{8}$$

where $\mu_i(\Omega) = -\frac{\int_\Omega Ae(u):q(\psi_i,\psi_i)dx}{\int_\Omega Ae(\psi_i)e(\psi_i)dx}$. Hence, the Lagrange function can be rewritten as

$$\boldsymbol{l(\Omega,u,v) = -\frac{\int_\Omega Ae(u):q(\psi_i,\psi_i)dx}{\int_\Omega Ae(\psi_i)e(\psi_i)dx} + \int_\Omega Ae(u)e(v)dx - \int_{\Gamma_N} p_0 v ds}, \boldsymbol{u,v \in U_{ad}}. \tag{9}$$

We should obtain the adjoint field $\boldsymbol{v}$. Because $(\boldsymbol{u,v})$ is a stationary point of the Lagrange function that satisfies the optimality conditions of the minimization problem, the partial derivatives of the Lagrange function with respect to $\boldsymbol{u}$ and $\boldsymbol{v}$ are separately expressed as

$$\left\langle \frac{\partial l}{\partial v}(\Omega,p,q),\vartheta \right\rangle = 0 = -\int_\Omega \vartheta \cdot div(Ae(p))dx + \int_{\Gamma_N} \vartheta \cdot (Ae(p) \cdot n - p_0)ds, \tag{10}$$

$$\left\langle \frac{\partial l}{\partial u}(\Omega,p,q),\vartheta \right\rangle = 0 = -\frac{\int_\Omega Ae(\vartheta):q(\psi_i,\psi_i)dx}{\int_\Omega Ae(\psi_i)e(\psi_i)dx} + \int_\Omega Ae(\vartheta)e(q)dx, \tag{11}$$

where $\boldsymbol{\vartheta}$ is the direction of $\boldsymbol{u,v}$, and, to simplify calculation, we set $\int_\Omega Ae(\psi_i')e(\psi_i')dx$ to 1 and define ψ_i' as $\frac{\psi_i}{\sqrt{\int_\Omega Ae(\psi_i)e(\psi_i)dx}}$. Equation (11) can be rewritten as

$$\left\langle \frac{\partial l}{\partial u}(\Omega,p,q),\vartheta \right\rangle = 0 = -\int_\Omega Ae(\vartheta):q(\psi_i',\psi_i')dx + \int_\Omega Ae(\vartheta)e(q)dx. \tag{12}$$

Because of equilibrium equation, Equation (10) is always equal to zero, with equation (12), we can select the adjoint field χ in Ω according to

$$\int_\Omega Ae(p)e(\chi)dx = \int_\Omega Ae(p):q(\psi_i',\psi_i')dx, \tag{13}$$

and can solve for it by the finite element method, and the shape derivative of Lagrange function $l(\Omega,p,q)$ can be obtained by using the concept of the shape derivative,

$$\begin{aligned} l'(\Omega,p,q) \cdot \theta = &-\int_{\Gamma_0} (Ae(p):q(\psi_i',\psi_i'))(\theta \cdot n)ds - \mu_i \int_{\Gamma_0} (Ae(\psi_i')e(\psi_i'))(\theta \cdot n)ds \\ &+ \int_{\Gamma_0} (Ae(p)e(\chi))(\theta \cdot n)ds. \end{aligned} \tag{14}$$

Because of the chain rule, the shape derivatives of the mathematical model in equation (7) can be expressed as

$$L'(\Omega,p,q) \cdot \theta = \frac{\sum_{i=1}^{n} e^{\rho_{KS}(\mu_1-\mu_i)} l'(\Omega,p,q) \cdot \theta}{\sum_{i=1}^{n} e^{\rho_{KS}(\mu_1-\mu_i)}}. \tag{15}$$

Satisfying the buckling-constraint of topology optimization problem. Lagrange function can be expressed as

$$L_1'(\Omega,u,q) \cdot \theta = \Lambda L'(\Omega,u,q) \cdot \theta + \int_{\Gamma_0} \theta \cdot n ds + \int_{\Gamma_N} \theta \cdot n ds. \tag{16}$$

For buckling-constraint topology optimization problems, according to the above form, the Lagrange multiplier Λ is

$$\Lambda^{k+1} = \begin{cases} \varpi\left[\boldsymbol{g}_{\lambda} - \boldsymbol{g}_{\lambda 1} + \boldsymbol{g}_{\lambda 1}\frac{\boldsymbol{k}}{\boldsymbol{n_r}}\right], & \boldsymbol{k} \le \boldsymbol{n_r}, \\ \Lambda^{\boldsymbol{k}} + \boldsymbol{\gamma}^{\boldsymbol{k}}\boldsymbol{g}_{\lambda}, & \boldsymbol{k} > \boldsymbol{n_r}, \end{cases} \tag{17}$$

where ϖ is a fixed parameter. γ^k will change during optimization as

$$\boldsymbol{\gamma}^{\boldsymbol{k+1}} = \mathbf{min}\left(\boldsymbol{\gamma}^{\boldsymbol{k}} + \Delta\boldsymbol{\gamma}, \boldsymbol{\gamma}_{\boldsymbol{max}}\right), \boldsymbol{k} > \boldsymbol{n_r}, \tag{18}$$

Method summary

The method for topology optimization problem with the buoyancy-weight ratios minimization can be summarized as follows. The flowchart of the algorithm is shown in Fig. 3.

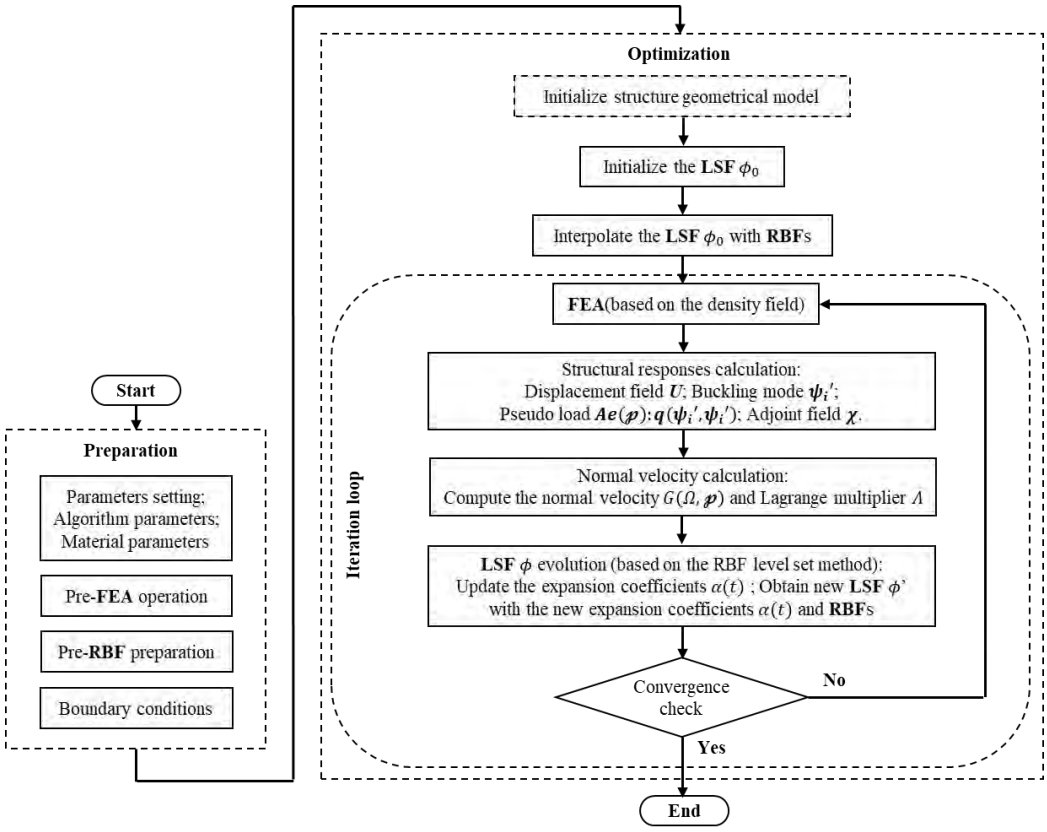


Figure 3: Flowchart of optimization procedure.

Preparation: Input the material and algorithm parameters. Prepare for finite element analysis by inputting information about the element numbers. Input a set of radial basis functions. Input the boundary conditions and load vector $\boldsymbol{F}$ of the structure.

Optimization: Initialize the initial structure and the level-set function ϕ_0 to define the initial structure contour and design domain. Fit the level-set function to obtain expansion coefficients $\alpha(t)$ of this set of radial basis functions.

Enter the iteration steps: Assemble the total stiffness matrix $\boldsymbol{K}$ of the current structure, obtain displacement field vector $\boldsymbol{U}$, obtain the first four orders of buckling mode vectors ψ_i by linear buckling analysis, and obtain adjoint field χ through finite element analysis. Compute term $L_1^{'}(\Omega, \boldsymbol{u}, \boldsymbol{q})$ at every nodal point in the normal velocity and obtain Lagrange multiplier Λ for the design constraints. Evolve the level-set function ϕ via the Hamilton–Jacobi equation, and update weight coefficients $\alpha(t)$ of the radial basis functions and obtain new level-set functions with new weight coefficients $\alpha(t)$ and the set of radial basis functions. Determine the convergence and whether the relative error of the objective function in the last nine adjacent

iterations is within reasonable ranges. If the convergence condition is not met, the iteration loop restarts from finite element analysis of the current structure.

Numerical examples and discussion

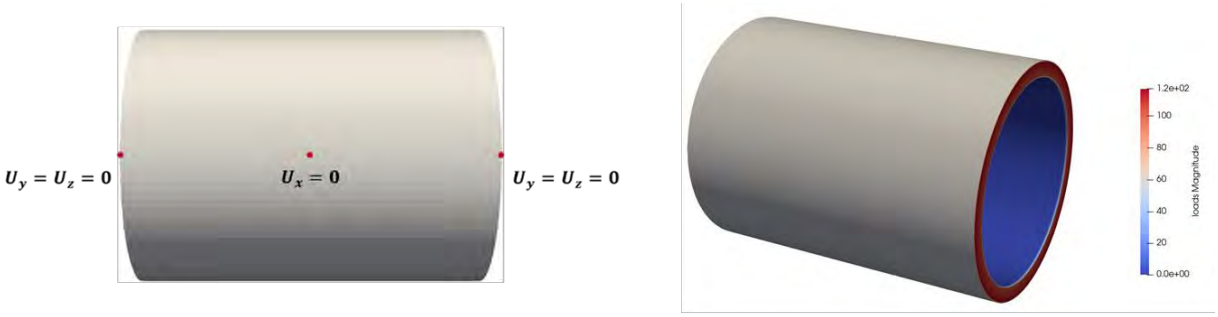
Cylindrical pressure hull is a common pressure hull used on large deep-sea submarines and deep-sea space stations. It has the performance of high space utilization, low fluid motion resistance, and convenient manufacturing. Therefore, due to the special geometric configuration, cylindrical pressure hulls have wide applications in the design of pressure hulls. In the design of underwater pressure hull, the structure is subjected to external hydrostatic pressure. The design area of the cylindrical pressure hull is set as follows: 200 mm in length, 10 mm in thickness, and 75 mm in radius. The required service depth for pressure hull is 1500 m. The geometric dimensions of the pressure hull are designed as follows: the flange height of the pressure hull is set to 10 mm, and the flange width is set to 2 mm, and the wall thickness is set to 3mm. In order to ensure the strength requirements of the underwater pressure hull, the wall thickness should not be less than 2.5 mm. Two ribs are set inside the pressure hull, and the two ribs are symmetrical along the middle surface of the pressure hull, with the distance of 66.7 mm from both ends. The geometric dimensions of the ribs are as follows: the length of the ribs is set to 10 mm, and the width is set to 14 mm. The design area, displacement boundary condition, and load boundary condition of the problem are shown in Fig. 4(a) and (b).

In this paper, titanium alloy is considered to build the hull because of the low density. The material parameters are described as follows:

Table 1: Material parameters

	Young's Modulus	**Poission ratio**	**Yielding strength**	**Density**
Value	1.12×10^5 Mpa	0.33	890 Mpa	4.5×10^3 g/mm^3

Fig. 5 shows the intermediate structure contours during optimization. Due the fact that the shape of pressure hull remains unchanged during the optimization process. Fig. 6 shows the convergence curves of volume fraction. Fig. 7 shows the convergence curve of the first four orders buckling factors and the aggregation function values of the pressure hull. Fig. 8 shows the convergence curve of maximal von-Mises stress value. It could be found that, on the one hand, the optimized structure has a smooth surface, and on the other hand, the rib section of the pressure hull changed a rectangle in the initial state to a "T" shaped rib, and the two ribs are chiral symmetric along the middle surface of the pressure hull. From Figure 6, it can be found that the buoyancy-weight ratio of the pressure hull has been improved by 19% through the proposed method. During the optimization process, it could be found that, through the iteration steps, changes of the critical buckling factors and Maximal values of von- Mise stress satisfied the requirement of 1500 m underwater environment.



(a) **(b)**

Figure 4: Cylindrical pressure hull example: (a) Design area, displacement boundary condition; (b) Load boundary condition.

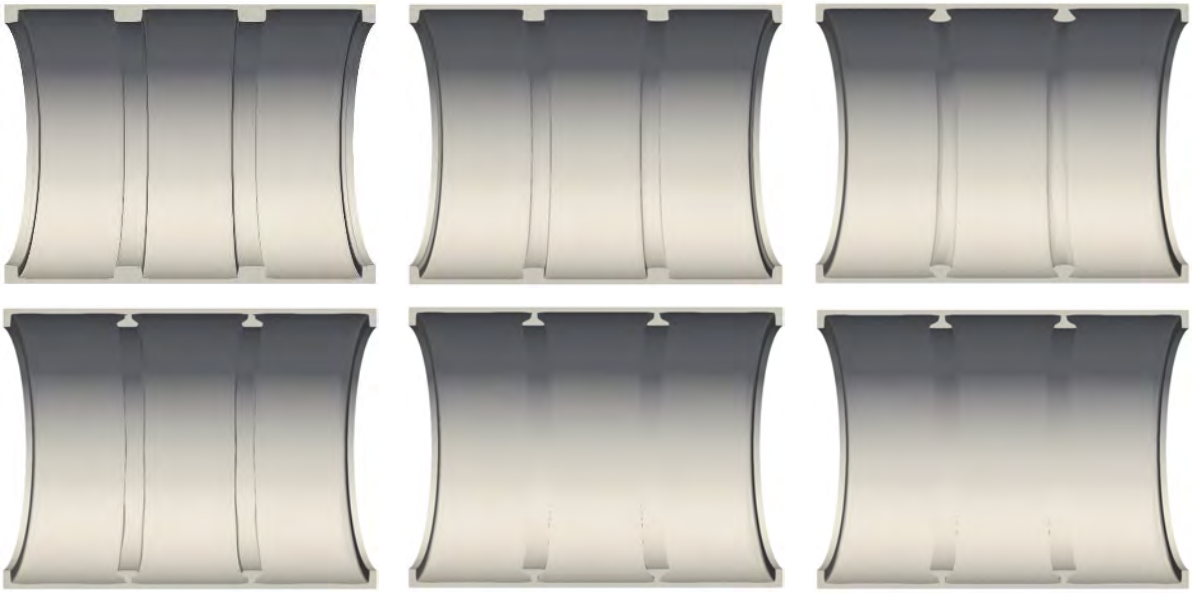


Figure 5: Intermediate structures of cylindrical pressure hull problem.

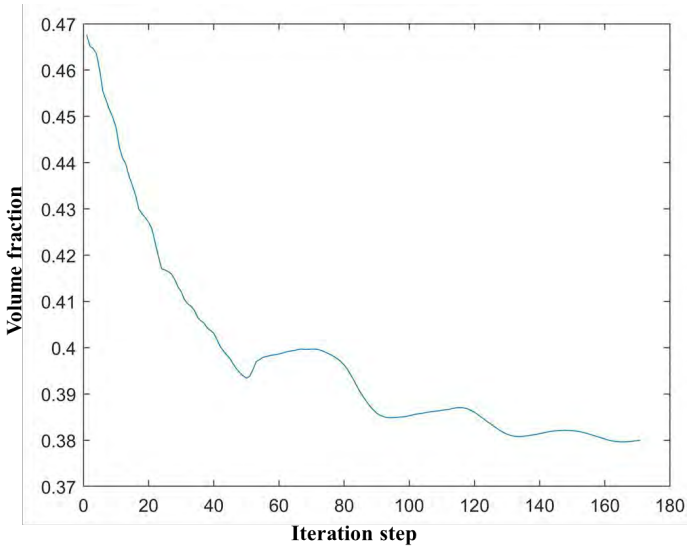


Figure 6: Convergence curves of the volume fraction.

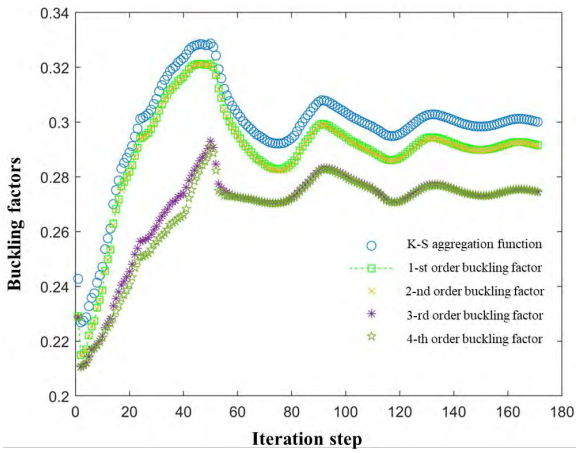


Figure 7: Convergence curves of buckling factors.

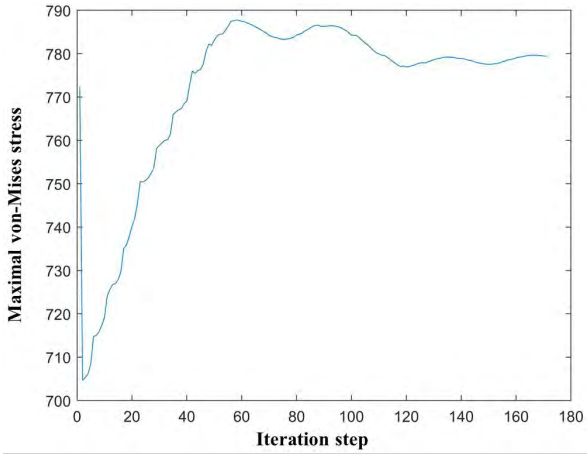
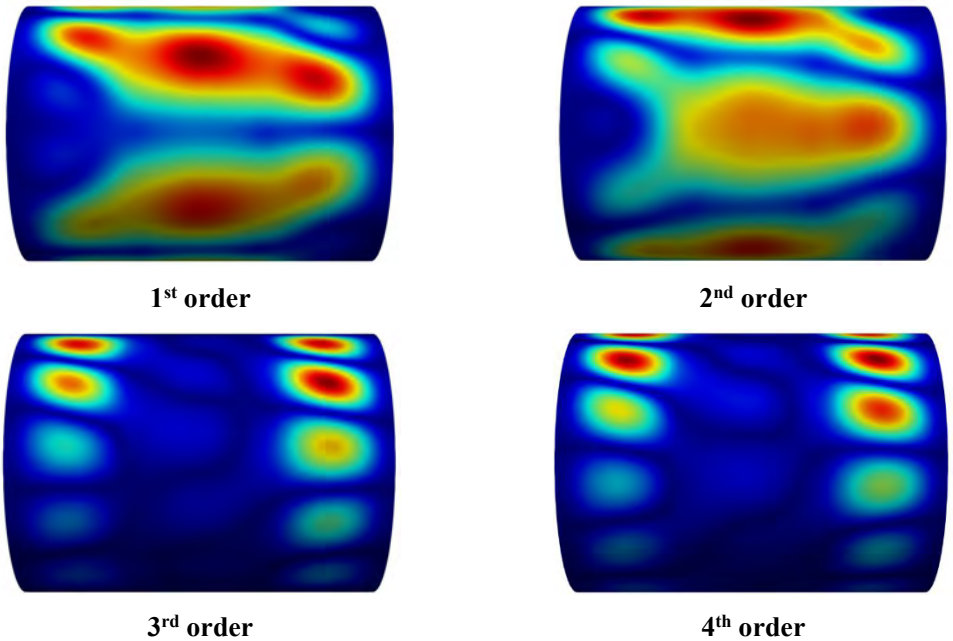


Figure 8: Convergence curves of the first four orders buckling factors and the aggregation function values.

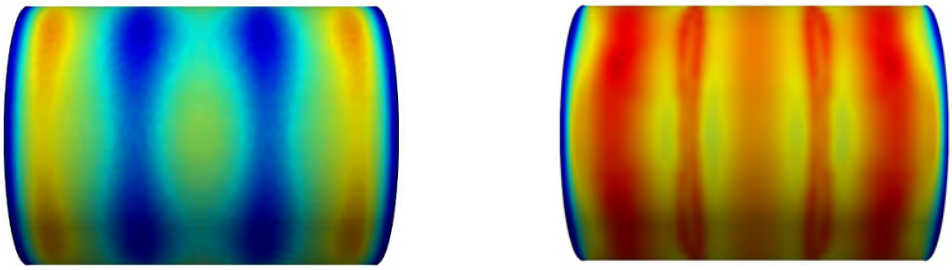
Fig. 9 shows the first four buckling-modes of the optimized structure. Fig. 10 shows the displacement field and von-Mises stress field cloud map of the final result from the outside of pressure hull, and Fig. 11 shows the fields of the inside. It can be observed that the deformation of the pressure hull is well-distributed, and the maximum deformation in the buckling mode is located near ends of the pressure hull. Though the given von-Mises stress field, the maximal value is mainly distributed at the wall of the pressure hull, and the value of von-Mises stress near the connection between the wall and the ring-ribs is far less than the yielding strength.



1st order **2nd order**

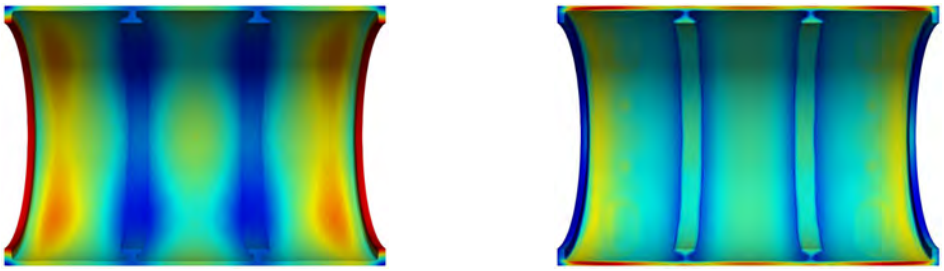
3rd order **4th order**

Figure 9: First four order buckling modes.



(a) (b)

Figure 10: Outside of pressure hull: (a) Displacement field, (b) von-Mises stress field.



(a) (b)

Figure 11: Inside of pressure hull: (a) Displacement field, (b) von-Mises stress field.

All in all, in this paper, the topology optimization method is proposed to optimize a pressure hull buoyancy-weight ratios minimization considering the critical buckling and yielding strength constraints, and the changes of each performance are shown. In the same case, the buoyancy-weight ratios of the pressure hull improved by 19% under the optimization process, and the ring-ribs has undergone a significant change in shape, which show an improving load capacity of the submersible to carry more batteries or sensors on board.

Conclusions

In this work, the optimal design of underwater pressure hull is studied by using the buoyancy-weight ratios minimization as the objective function, the critical buckling factor and the yield strength as the design constraint. Topology optimization of cylindrical underwater pressure hull problem is solved effectively. Compared to the regular design, the shapes of ring-ribs and the spacings between ribs could be changed during the optimization process. The proposed method provides greater design freedom for the design of underwater pressure hull, which is helpful to improve the stability performance of pressure hull.

Acknowledgements

The authors would like to deeply appreciate the support from the National Key Research and Development Program of China (No. 2021YFC28026004, No. 2021YFC28026000), National Natural Sciences Foundation of China (U2067220, 51779139) and Top Young Talents of Ten Thousand Talents Plan. The authors also thank Professor Peng Wei et al. for providing the 88-line MATLAB source code for the parameterized level-set method based on topology optimization using radial basis functions.

References

[1] Yang, M., Wang, Y., Chen, Y., et al. (2022) Data-driven optimization design of a novel pressure hull for AUV, Ocean Eng. 257, 111562.

[2] Zhang, X., Sun, J., Wu, R., et al. (2023) Buckling performance of inner corrugated pressure shells under external hydrostatic pressure, Ocean Eng. 288, 115963.

[3] Lindgaard, E., Dahl, J. (2013) On compliance and buckling objective functions in topology optimization of snap-through problems, Struct. Multidiscip. Optim. 47, 409-421.
[4] Ferrari, F., Sigmund, O., Guest, J.K. (2021) Topology optimization with linearized buckling criteria in 250 lines of MATLAB, Struct. Multidiscip. Optim. 63, 3045-3066.
[5] Zhang, W., Jiu, L., Meng, L. (2022) Buckling-constrained topology optimization using feature-driven optimization method, Struct. Multidiscip. Optim. 65, 37.
[6] Xu, T., Lin, X., Xie, Y.M. (2023) Bi-directional evolutionary structural optimization with buckling constraints, Struct. Multidiscip. Optim. 66, 67.
[7] Xu, T., Huang, X., Lin, X., Xie, Y. M. (2023) Topology optimization for maximizing buckling strength using a linear material model, Comput. Methods Appl. Mech. Engrg. 417A, 116437.
[8] Deng, H. (2024) A novel discrete adjoint-based level set topology optimization method in B-spline space, Optim. Engrg. DOI: 10.1007/s11081-023-09851-7.
[9] Wang, S.Y., Lim, K.M., Khoo, B.C. et al. (2007) An extended level set method for shape and topology optimization, J. Comput. Phys. 221(1), 395-421.
[10] Jiang, Y., Zhao, M. (2020) Topology optimization under design-dependent loads with the parameterized level-set method based on radial-basis functions, Comput. Methods Appl. Mech. Engrg. 369, 113235.
[11] Jiang, Y., Zhao, M. (2021) Stress-based topology optimization with the parameterized level-set method based on radial basis functions, Struct. Multidiscip. Optim. 65, 224.
[12] Otomori, M., Yamda, T., Izui, K., Nishiwaki, S. (2015) Matlab code for a level set-based topology optimization method using a reaction diffusion equation, Struct. Multidiscip. Optim. 51, 1159-1172.

Development of a system for classifying J-core and UKHardcore music genres using music2vec

***Nanase Kishi[1], †Ryuji Shioya[2], Yasushi Nakabayashi[2]**

[1] Graduate School of Information Sciences and Arts, Toyo University, Japan.
[2] Faculty of Information Sciences and Arts, Toyo University, Japan.

*Presenting author: s3b102300052@toyo.jp
†Corresponding author: shioya@toyo.jp

Abstract

In recent years, advances in AI have led to a rapid increase in tools such as natural language generation and image generation. AI technologies in the music field include Izotope's Ozone11 for automatic mastering and Mubert's tool for automatically generating music from text. However, AI-based music analysis tools for classification purposes are not yet widespread.

This research aims to use AI and machine learning to classify music genres and assist artists. By objectively analyzing their own music, artists can better understand the characteristics of their music and incorporate popular trends. Additionally, they can also evaluate their own work, set goals, and drive continuous improvement. In this study, we use Music2vec[1] as a method to convert music into vector representations for classification. We also explore previous work using Soundnet[2] and CRNN for music analysis. The definition of musical genre establishes its two defining criteria: classification based on performance form and musical structure and classification based on historical description.

In this study, we conducted experiments using a dataset of UK Hardcore and J-core genres to train machine learning models. The inputs consisted of 30-second tracks, which were used as Mel spectrograms. We trained the models on 180 songs from each genre and compared of the models with and without stride. The model trained without stride achieved 71% accuracy, while the model trained with stride achieved 80% accuracy. This shows that using stride to expand the dataset can lead to higher accuracy.

This research also demonstrates new possibilities for artists, showing how AI can provide insights into their music that they may not have been aware of. By using these insights in self-promotion, artists can reach an audience that more closely matches with their intended market.

Keywords: Music2vec, Mel Spectrogram, Soundnet.

Introduction

In recent years, there has been remarkable progress in AI, with increasing opportunities for its application becoming commonplace. Examples include ChatGPT by OpenAI [1] and Stable Diffusion by Stability AI [2]. ChatGPT is capable of generating text, while Stability AI can produce images and artistic works. Moreover, various products utilizing AI for music have emerged. For instance, Ozone 11 by Izotope [3] features AI-driven suggestions for mastering tasks. Additionally, Mubert by Mubert [4] offers a tool for automatic music generation from text. Thus, AI is believed to provide diverse approaches from a third-party perspective in

supporting creative works.
However, tools for music analysis using AI have yet to be widely adopted. Analyzing one's own compositions with AI to determine differences compared to currently popular songs could provide valuable support to artists.
The objective of this research is to leverage AI and machine learning for music classification to provide support for artists. Music genres are defined by humans and categorized based on their distinct characteristics [5]. Music classification by AI offers artists an objective perspective on their own compositions, facilitating the incorporation of features and characteristics from popular songs into their own work. Understanding the differences between composed songs and popular music enables artists to evaluate themselves, set improvement goals, and facilitate continuous growth. Further subdivision of categorized music genres allows for distinguishing the specific characteristics of each song.

Previous Studies

1. Music2vec [6]

One of the methods for music classification using AI is Music2vec. Music2vec, developed by Rajat Hebbar, converts music into vector representations. This technique applies features from text-based information retrieval models. The dataset utilized is the Free Music Archive (FMA), consisting of a large collection of 106,574 tracks from 16,341 artists and 14,854 albums, organized into a hierarchical classification of 161 genres. Mel spectrograms are used as input information. While spectrograms are commonly used for visualizing audio in a 2D representation of time and frequency axes, mel spectrograms are employed for faster computational processing. Figure 1 depicts a mel spectrogram of electronic music used in the study. By converting this mel spectrogram into vector representations and conducting learning, the test set achieved an accuracy of 43%.

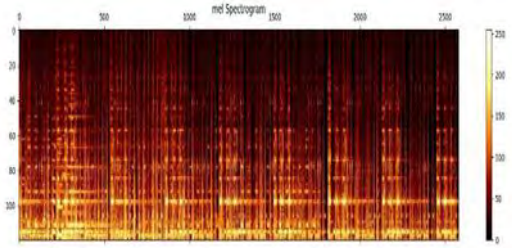


Figure 1. Mel Spectrogram of Electronic Music

2. Soundnet

Soundnet is a deep learning model for audio recognition in machine learning [7]. It utilizes Convolutional Neural Networks (CNNs), a type of deep learning, to learn features of audio. According to research, Soundnet learns from large amounts of unlabeled video data and performs classification of acoustic scenes and objects. Video data contains both visual and audio information, and Soundnet learns audio features using visual recognition models. Figure 2 displays labels recognized by AI from audio, including objects and landscapes, as reported in the study.

Figure 2. Image Displaying Labels Recognized by Soundnet for Objects and Landscapes from Audio

3. CRNN(Convolutional Recurrent Neural Network)

CRNN combines CNN (Convolutional Neural Network) and RNN (Recurrent Neural Network). It is used to model continuous data such as audio signals and word sequences. CNN, known as Convolutional Neural Network, is utilized for processing mel spectrogram images. Then, the music's time-series data is handled by RNN, a type of recurrent neural network. According to the research by Keunwoo Choi et al. [8] which utilized CRNN for music classification, it reported higher effectiveness in music tagging compared to three other CNN models.

Rerearch Methodology

1. Three Types of Classification

There are three types of classification to be conducted. Firstly, classification by genre, which solely categorizes music genres such as rock and jazz to examine the extent to which genres can be further subdivided. Even within rock, there exist closely related genres like metal and punk, which can be further differentiated. This aims to investigate whether AI can classify genres that are difficult even for humans to distinguish. Secondly, classification by country, which categorizes songs based on the nationality of the composing artist. This investigates whether classification is possible based on the artist's country of origin, even for music of the same genre. Lastly, classification of popular songs in each country. Although popular songs vary by country, this classification aims to train the model on the hit charts of each country to classify popular songs.

2. Definition of Music Genres

Defining each music genre is necessary when preparing the dataset. However, to date, there is no universal definition of music genres. For example, the genre of rock generally emphasizes electric guitar sounds, and it may include instruments such as bass and drums in a band setup. Similarly, the genre of metal primarily features distorted electric guitars, bass, and drums, which are also present in rock. The difference between rock and metal is believed to lie in metal's emphasis on low-frequency sounds, although there isn't a clear distinction in low frequencies. Furthermore, there are rock songs that emphasize low frequencies as well. In many cases, music genres lack clear classification boundaries. However, by considering the following two items, it is possible to differentiate unclear music genres when defining music genres [9]. Firstly, classification based on performance forms and musical structures. For example, in classical music, there are symphonies and concertos, while in popular music, there are bands, solo performances, and instrumentals. Secondly, classification based on historical descriptions, such as the classification of classical music into Baroque, Classical, Romantic, and Contemporary periods. By considering these two items, it is possible to prepare a dataset for music genre classification.

Experiment

1. Genre of Analyzed Songs

In this study, the analyzed songs focus on the genre of Hardcore, one of the subgenres of Electronic Dance Music (EDM). Hardcore emerged in the early 1990s in countries such as the Netherlands, Belgium, and Germany [10]. In Japan, unique derived genres like Jcore and JHardcore have been established, while in the UK, it is known as UKHardcore. The definition

of Hardcore for the two countries is "Hardcore in general released by labels and track makers within each country" [11].

2. Experimental Method

The experiment involves converting music into vector representations and conducting classification using Music2vec and Soundnet. Referring to the model by Rajat Hebbar and KMASAHIRO [12]. As input, 30-second music data and corresponding country vectors were loaded for training. The output was assigning country labels to data every 30 seconds. Python was used as the programming language. The file format was WAV (Waveform Audio File Format), with a playback time of 30 seconds and a sampling rate of 22050Hz. When preparing the model, a function to construct Soundnet using pre-trained weights was used.

3. Training Data

Two types of input data were prepared: three songs and ten songs for each country. Since each song is from two different countries, there are a total of six songs and twenty songs, respectively. Each song was divided into 30-second segments to increase the amount of data. Two methods of segmentation are shown in Figure 3. One method divides the song into 30-second segments without striding. The other method strides within one song to divide it into 30-second segments, aiming to increase the training data. It is ensured that one piece of data does not span multiple songs.

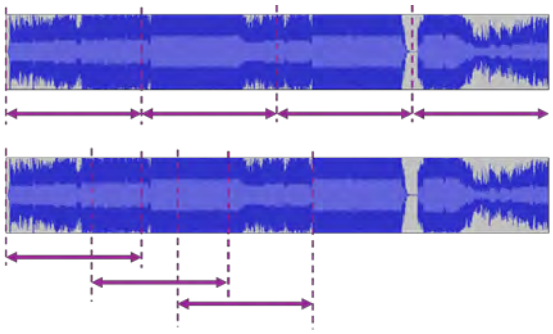


Figure 3. Above: Segmentation without striding Below: Segmentation with striding.

4. Test Data

Test data consisted of five Hardcore songs produced by Japanese artists and five produced by British artists. Each song was divided into 50 datasets, resulting in a total of 100 datasets. These 100 datasets were evaluated using Music2vec to determine the accuracy. Additionally, the test songs were not included in the training data.

5. Results of Training Data with 40 Datasets (3 Songs)

Training was conducted with six songs divided into 40 datasets without striding, and the results of testing 100 datasets are shown in Table 1. The accuracy was 50%. As the predictions are column-wise, all 100 datasets were classified as Japanese Hardcore.

Table 1. Results of Training with 40 datasets (3 songs)

	Jcore	UKHardcore
Jcore	50	0
UKHardcore	50	0

6. Results with 120 datasets (20 songs) for Training Data

Next, training was conducted on twenty songs divided into 120 datasets without striding, and the results of testing are shown in Table 2. The accuracy was 74%, which was higher than the accuracy achieved with six songs.

Table 2. Results of Training with 120 datasets (20 songs)

	Jcore	UKHardcore

Jcore	45	5
UKHardcore	21	29

7. Results with 360 datasets (3 songs) for Training Data

Similarly, training was conducted on twenty songs divided into 360 datasets, and the results of testing are shown in Table 4. The test resulted in 77% accuracy. All datasets without striding achieved higher accuracy than those without striding.

Table 3. Results of Training with 360 datasets (3 songs)

	Jcore	UKHardcore
Jcore	40	10
UKHardcore	12	38

8. Results with 360 datasets (20 songs) for Training Data

Similarly, training was conducted on twenty songs divided into 360 datasets, and the results of testing are shown in Table 4. The test resulted in 77% accuracy. All datasets without striding achieved higher accuracy than those without striding.

Table 4. Results of Training with 360 datasets (20 songs)

	Jcore	UKHardcore
Jcore	47	3
UKHardcore	20	30

Consideration

In this experiment, we observed differences in AI classification accuracy based on the quantity of training data. Increasing the number of input songs or expanding the volume of training data can lead to improved accuracy. Additionally, considering the maximum accuracy of 78%, it can be inferred that the AI has grasped the characteristics of Japanese and British Hardcore. Below, we discuss the differences between Japanese and British Hardcore concerning the key and BPM (Beats Per Minute) used in the training data, as summarized in Table 5, and illustrated in Figure 4.

According to Table 5, Japanese Hardcore has an average BPM of 186.7, with 4 out of 10 songs in major keys, indicating a faster tempo and brighter compositions compared to British Hardcore. Conversely, British Hardcore has an average BPM of 165.5, with 7 out of 10 songs in minor keys, suggesting a slower tempo and darker compositions compared to Japanese Hardcore. These differences are likely influenced by each country's market. Japanese Hardcore is primarily produced for Japan's music gaming market. This can be attributed to the establishment of Jcore in the early 2000s [13], coinciding with the late 1998 music gaming boom, which introduced Hardcore into the music gaming market. Consequently, Jcore is expected to be sold at events related to music gaming, leading to an increase in compositions targeting music gaming enthusiasts. The higher average BPM in Japanese Hardcore may be due to setting higher difficulty levels for fingertip-based gaming rather than dance-based music typically played in clubs, compared to British Hardcore. On the other hand, British Hardcore targets primarily overseas club markets. This can be traced back to Hardcore Techno, the predecessor of UKHardcore, being used as EDM in clubs and festivals. As a result, UKHardcore is expected to be sold as dance music, leading to an increase in compositions used by DJs and performance artists. Considering these observations, the reason

Music2vec demonstrated high accuracy may be attributed to considering differences in BPM and key signatures of the compositions.

Table 5. Key and BPM

BPM	Jcore	UKHardcore
Average	186.7	165.5
Minimum	165	153
Maximum	222	170
Major Key	4tracks	3tracks
Minor Key	6tracks	7tracks

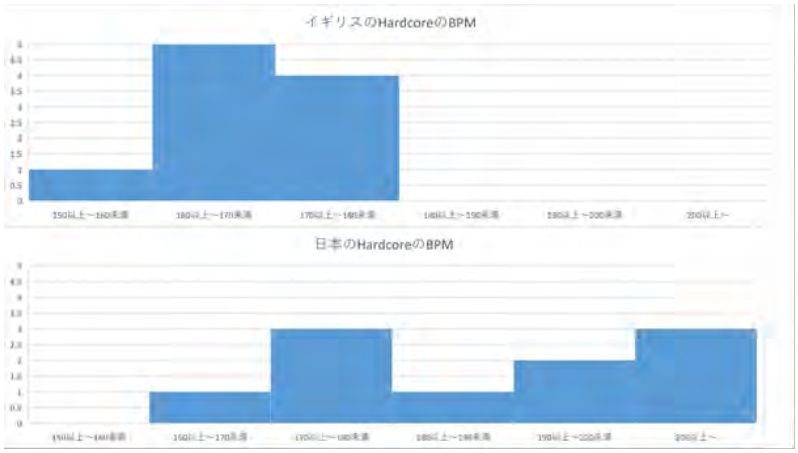


Figure 4. Top: BPM of Japanese Hardcore, Bottom: BPM of British Hardcore

Conclusion

In this experiment, we utilized Music2vec for music classification. The results showed that when training with 40 datasets without sliding 6 songs, the accuracy was 50%. When training with 120 datasets without sliding 20 songs, the accuracy improved to 74%. Training with 360 datasets by sliding 6 songs resulted in an accuracy of 78%, and training with 360 datasets by sliding 20 songs yielded an accuracy of 77%. These findings indicate that increasing the original number of songs in the training data and augmenting the data by splitting songs contribute to improving the accuracy of Music2vec.

Furthermore, the high accuracy can be attributed to the characteristics of the Japanese and British music markets. The Japanese Hardcore market includes compositions targeted at music gaming enthusiasts due to the prevalence of music gaming-related compositions and events, while the British Hardcore market targets events such as clubs and festivals, owing to the history of Hardcore Techno being used in DJ-centric contexts.

Future research directions include standardizing the data extracted from the songs used for training and testing. Currently, each song is divided into 30-second segments, so each data point is classified based on the country, requiring a mechanism to assign a single label per song. Additionally, classifying songs based on popularity provides another research avenue aimed at revealing the preferences of music consumers in Japan and the UK. We plan to proceed with classification using Spotify API's popularity metric, as it allows us to understand real-time popularity rather than cumulative values like play counts.

The potential of the system developed in this study lies in its ability to provide artists with a multifaceted perspective by classifying their compositions using AI, thereby contributing to the demand for internationalized content.

Acknoledgement

This work was supported by Cabinet Office, Government of Japan,Cross-ministerial Moonshot Agriculture, Forestry and Fisheries Research and Development Program, "Technologies for Smart Bio-industry and Agriculture" (funding agency: Bio-oriented Technology Research Advancement Institution)

References

[1] OpenAI,ChatGPT,2024. https://openai.com/
[2] Stable AI ,Stable Diffusion,2024. https://ja.stability.ai/stable-diffusion
[3] Izotope,Ozone11,2024. https://www.izotope.jp/jp/products/ozone-11/#
[4] Mubert,Mubert,2024. https://mubert.com/
[5] Chillara, Snigdha, et al. "Music genre classification using machine learning algorithms: a comparison." Int Res J Eng Technol 6.5 (2019): 851-858.
[6] Ashish Bharadwaj Srinivasa, Rajat Hebbar “music2vec: Generating Vector Embeddings for Genre-Classification Task”(2017), https://medium.com/@rajatheb/music2vec-generating-vector-embedding-for-genre-classification-task-411187a20820
[7] Aytar, Yusuf, Carl Vondrick, and Antonio Torralba. "Soundnet: Learning sound representations from unlabeled video." Advances in neural information processing systems 29 (2016).
[8] Choi, Keunwoo, et al. "Convolutional recurrent neural networks for music classification." 2017 IEEE International conference on acoustics, speech and signal processing (ICASSP). IEEE, 2017.
[9] What is the genre of music?,JUN,2021. https://note.com/junjunjunpiano/n/nc57d89fed91f
[10] Aggressive Dance Music. Masterpieces of Hardcore Techno, RAG MUSIC Editorial Department,2023. https://techno.studiorag.com/hardcore-techno-songs?disp=more
[11] Ny4nne's Thoughts on J-CORE Genre Definitions, Ny4nne, 2022. https://note.com/nakisunachan/n/n4d2e9341e469
[12] Music Genre Classification Model Using music2vec, KMASAHIRO,2021. https://qiita.com/KMASAHIRO/items/cae4dfb0657eec4a2dca
[13] J-CORE Cultural Revolution, Rough Sketch,2007. https://data.technorch.com/data/page/gbn93.html

Simulation of complex flow with a hovering dragonfly by multi-direct forcing immersed boundary method

*** Dongli Han [1], † Xing Shi [1], and Yao Zheng [1]**

[1] School of Aeronautics and Astronautics, Zhejiang University, Hangzhou, Zhejiang, 310027, China.

*Presenting author: 22024031@zju.edn.cn
†Corresponding author: shix@zju.edu.cn

Abstract

The aerodynamic mechanism of a flapping-wing flighter is very complicated, because of the rapid wing movement and the inherent complex vortex dynamics. In this study, the immersed boundary method, with prediction–correction multi-direct forcing scheme and smoothed discrete delta function, was proposed using the open source toolbox OpenFOAM version 7. In the approach, the interaction between fluids and moving rigid boundaries is realized by adding a volume forces source term to the incompressible Navier-Stokes equation. The multi-direct forcing scheme makes the boundaries better meet the no-slip boundary condition, and further, the smoothing technique for the discrete delta function was applied to suppress the non-physical oscillations of volume force.

In contrast to most common insects, a dragonfly has two pairs of wings in tandem, and the interactions between forewings and hindwings play an important role in aerodynamic performance. In this study, the single-wing hover with no interaction and the tandem-wing hover with interaction are simulated and compared, the interaction effect reduces the vertical forces on the forewings and hindwings by 14% and 16%, respectively. The above effects are mainly due to the interactions among the vortex structures of tandem wings, which change the pressure distributions of the wing surfaces and the flow field near the wings. In the hovering mode, the average vertical force is equivalent to the gravity of a dragonfly, and the average thrust force is approximately zero. This study reveals the flow characteristics of dragonfly flapping wings, which provides a theoretical basis for designing a series of flapping wing robots.

Keywords: multi-direct forcing scheme, smoothed discrete delta function, immersed boundary method, hovering dragonfly, tandem-wing interactions

1. Introduction

Over millions of years of evolution, insects have evolved the most adaptable and controllable flight systems in the world. They can achieve various maneuvers such as hovering, zooming, jerking, rapid acceleration and turning, even taking off backward, flying sideways, and landing upside down. Because of their small size, the relative speed of the wings is very small, so the insects require enough lift coefficient to achieve the above-mentioned flight. How they produce high lift under a low Reynolds number is a problem that both fluid mechanics and biologists are very concerned about. Studying the flying principles of insects, especially the high-lift mechanism, can guide the design of micro-aircraft. In addition, biologists need to understand the impact of the generation of aerodynamic forces and energy consumption in flight on the ecology, physiology, and evolution of insects.

In the past nearly 100 years, researchers have been devoted to exploring the flight mechanism of insects. Initially, researchers applied the quasi-steady analysis method to assess the

aerodynamic characteristics of insects, but it failed to predict the high-lift mechanism in insect hovering. Since the 1980s, with the advancement of experimental observation methods and the development of Computational Fluid Dynamics (CFD), researchers have begun to study flapping-wing aerodynamics from an unsteady perspective and have achieved a series of outstanding results. In terms of the experimental research, Ellington et al[1] used smoke flow visualization to study the airflow around both real and a hovering mechanical model of the hawkmoth Manduca sexta, and James et al[2] used two-dimensional digital particle image velocimetry (DPIV) to visualize the flow patterns around the flapping wing of a dynamically scaled robot for a series of reciprocating strokes starting from rest. Tien et al[3] employed the bubble visualization and force sensor to measure unsteady forces and visualize 3D vortices around a beetle-like flapping-wing model in hovering flight. In terms of Computational Fluid Dynamics, Wang et al[4] solve the Navier-Stokes equation in elliptic coordinates to simulate the flow field of a two-dimensional flapping wing in hover mode. Gong et al[5] developed a general large-scale parallel solver using Immersed Boundary-Lattice Boltzmann Method (IB-LBM) to explore the three-dimensional bionic flapping-wing flow mechanism. Kang et al[6] explored the effect of chordwise, spanwise, and isotropic flexibility on the force generation and propulsive efficiency of flapping wings. Lua et al[7] reported on their investigation of the influences of pitching phase angle and amplitude on the aerodynamics of a two-dimensional flapping wing that executes a simple harmonic motion in hovering mode.

In nature, dragonflies are one of the very few insects that can independently control the movement of their four wings, which makes dragonflies have excellent flight performance and can cope with complex flight conditions and achieve various difficult maneuvers such as hovering side flying, reverse flying, and gliding. Therefore, dragonflies have become one of the main biological templates for the study of bionic ornithopter aircraft.

In computational fluid dynamics (CFD), most of the flow problems can be simulated by body-fitted mesh. However, the generation of complex geometric body-fitted mesh is difficult, and the moving boundary problem requires additional processing methods such as moving mesh and overset mesh technology. The immersed boundary method has been developed to overcome these shortcomings, which was originally proposed by Peskin[8] to simulate cardiac mechanics and related blood flow in 1972. The solid boundary conditions of complex flow problems are realized on static background grids with constant topology by adding source term to the governing equation or modifying the fluid velocity near the boundary. In recent years, the immersed boundary method has been fully developed and is widely used in the fluid-solid coupling problem of flexible plastic materials or rigid bodies, and has derived many different forms, for example, the continuous forcing approach, the discrete forcing approach, and the ghost cell approach[9].

This study aims to investigate the effects of tandem-wing interaction on the aerodynamic performance of a three-dimensional hovering dragonfly. To complete the numerical simulation of rigid body incompressible flow problems, an improved pressure implicit splitting operator (PISO) solver is proposed in the open source toolbox OpenFOAM version 7, which combines the direct force immersed boundary method. The immersed boundary method introduces numerical inaccuracies for it does not enforce boundary conditions directly, notably at the solid-fluid interface where the actual boundary velocity may deviate from the expected value due to mesh cutting or interpolation error. To counteract this error and bolster the calculation's precision and convergence velocity, we have integrated a multi-direct forcing scheme into our approach. Furthermore, to quell the numerical oscillations arising from the motion of objects in the volume force, we have applied a smoothing technique to the original interpolation function.

The outline of this paper is the following: In Section 2, we introduce the computational model and kinematic parameters of the dragonfly and present the immersed boundary method to implement the simulation. The verification work of the algorithm and grid independence are

detailed in Section 3. In Section 4, the aerodynamic performance and flow field structure of the tandem wing are compared with that of the single wing. Finally, conclusions are made in Section 5, including some perspectives for future investigations.

2. Physical model and Numerical method

Dragonfly model and kinematics of the wing

In the past decades of research, researchers have modeled the movements of many insects. To simplify the model, we use a pair of rigid plates as shown in Figure 1 to approximate the wings of a dragonfly. The shape of the wings and the position of the torsion axis are obtained from the measurements of real dragonflies [10]. According to his measurements, the dragonfly has a mass of about 754 mg, a forewing length of 4.74 cm with a mean chord length of 0.81 cm, and a hindwing length of 4.60 cm with a mean span length of 1.10 cm. The radius of the second moment of the forewing area is denoted by r_2:

$$r_2 = \int_{S_f} r^2 dS_f / S_f \tag{1.1}$$

where r is radial distance and S_f is the area of the forewing); r_2=0.61R (R is the wing length). The kinematic model and related parameters of Sun and Lan[11] are considered, which hovers with an inclined stroke plane. To accurately describe the movement characteristics of dragonfly wings, we defined the following angles: firstly, the stroke plane angle β, namely the angle between the stroke plane and the horizontal plane; second, the flapping angle ϕ, namely the angle between the torsion axis on the flapping plane and the intersection line between the stroke plane and the horizontal plane, the third angle is the feathering(or pitching) angle θ, which is defined as the angle between the chord line of the wing and its projection on the stroke plane. The last one is the deviation(coning angle in Azuma's research[12]) angle ψ, represents the angle between the torsion axis and its projection on the stroke plane. Figure 2 shows the forewing, the body of the dragonfly model, and the four angles described above to describe the movement of the dragonfly. The axis OXYZ is the inertial coordinate system, and the origin of the coordinates is defined at the intersection of the torsion axis of the front wing and the torso of the dragonfly. The X-axis is parallel to the torso, the Y-axis is vertical and upward, and the Z-axis is obtained by the right-hand rule.

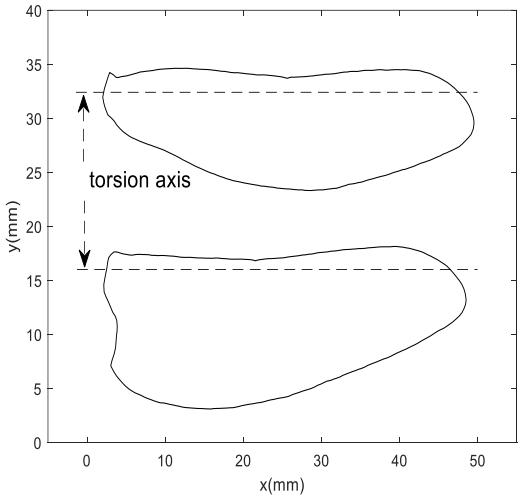


Figure 1. Wing geometry and the position of the torsion axis

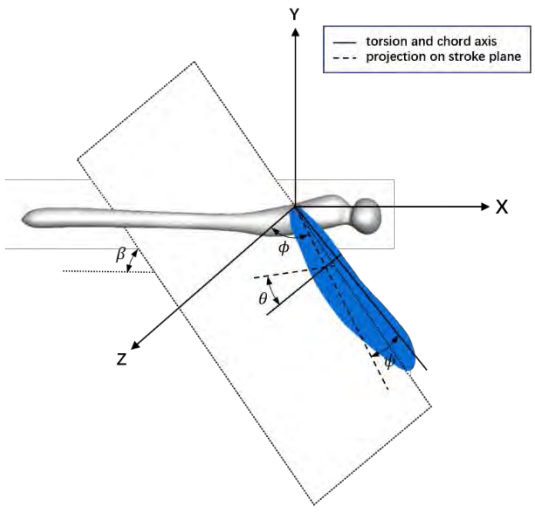


Figure 2. Sketches of the model dragonfly and the flapping motion

According to the observation results of Azuma[12], and Peng[13], the deviation angle of the forewing and hindwing of dragonflies is 8° and -2°, respectively. In recent studies, to further simplify the dragonfly movement model, the influence of the deviation angle is usually ignored, so the deviation angle is set to 0° in this paper. Therefore, the dragonfly's motion mode is simplified into two parts: translation on the stroke plane and pitch (or flip) rotation around the torsion axis near the end of a half-stroke and at the beginning of the following half-stroke. In the research of Sun[11], the translation velocity u_t is defined as the flapping velocity at r_2 and is given by:

$$u_t = 0.5\pi U \sin(2\pi\tau/\tau_c + \gamma) \tag{1.2}$$

Where the U is the reference velocity, $\tau = tU/c$ (t is the time and c is the mean chord length of the forewing), and τ_c is the non-dimensional time. The non-dimensional period of the flapping cycle, γ is the phase angle of the translation of the wing (180° for the forewing, 0° for the hindwing), the reference velocity U is calculated by:

$$U = 2\Phi n r_2 \tag{1.3}$$

Where the $\Phi = 69°$ is flapping angle amplitude, $n = 36Hz$ is flapping frequency. the reference velocity $U = 2.5m/s$. Based on the above data and the kinematic viscosity of air at room temperature υ, the dimensionless parameter Reynolds number can be calculated by the formula $\mathrm{Re} = Uc/\upsilon \approx 1350$. Through the relationship between the flapping angular velocity and the translation velocity $\dot{\phi} = u_t/r_2$, we finally get that the flapping angle changes with time as follows:

$$\phi = \bar{\Phi} - 0.5\Phi\cos(2\pi nt + \gamma) \tag{1.4}$$

$\bar{\Phi}$ is the average of the flapping angle in a period, for the forewings $\overline{\Phi_f} = 5.5°$, and for the hindwing $\overline{\Phi_h} = 17.5°$[14].

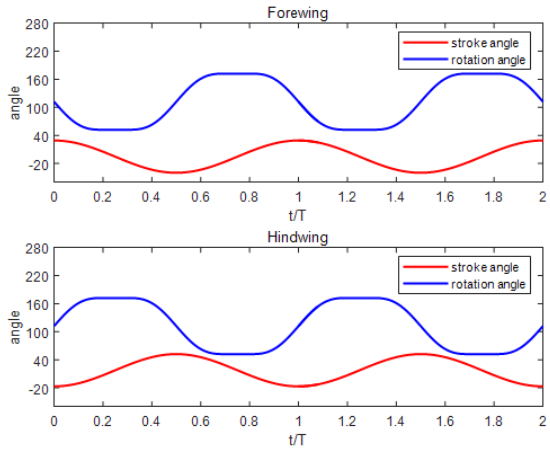


Figure 3. Kinematics of forewing and hindwing

In Sun's study, the pitch motion is obtained by defining the change law of the angle of attack α. An assumption is introduced here, that is, α remains constant in the middle of the upstroke and downstroke, and the angular velocity of the angle of attack during the stroke conversion is as follows:

$$\dot{\alpha} = 0.5c\dot{\alpha}_0\{1 - \cos[2\pi(\tau - \tau_r)/\Delta\tau_r]\}/U \tag{1.5}$$

Where $\dot{\alpha}_0$ is a constant determined by the fixed angle of attack in the middle part of the stroke ($\alpha_u = 172°$ for upstroke and $\alpha_d = 52°$ for downstroke) and the dimensionless time $\Delta\tau_r = 3.36$ that the reversal motion lasts, τ_r is the start time of the pitch motion. Under the assumption that the deviation angle is 0°, the definition of pitch angle and attack angle is consistent. Therefore, the variation rule of the pitch angle is given as:

$$\theta = \begin{cases} \dfrac{\alpha_u + \alpha_d}{2} + \dfrac{(\alpha_d - \alpha_u)t}{2\Delta T} - \dfrac{(\alpha_d - \alpha_u)}{2\pi}\sin\left(\dfrac{2\pi\left(t - \frac{\Delta T}{2}\right)}{\Delta T}\right) & -\dfrac{\Delta T}{2} \le t < \dfrac{\Delta T}{2} \\ \alpha_d & \dfrac{\Delta T}{2} \le t < \dfrac{1-\Delta T}{2} \\ \dfrac{(1+\Delta T)\alpha_d - (1+\Delta T)\alpha_u}{2\Delta T} + \dfrac{(\alpha_d - \alpha_u)t}{2\Delta T} + \dfrac{(\alpha_d - \alpha_u)}{2\pi}\sin\left(\dfrac{2\pi\left(t - \left(\frac{1-\Delta T}{2}\right)\right)}{\Delta T}\right) & \dfrac{1-\Delta T}{2} \le t < \dfrac{1+\Delta T}{2} \\ \alpha_u & \dfrac{1+\Delta T}{2} \le t < 1 - \dfrac{\Delta T}{2} \end{cases} \quad (1.6)$$

Where $\Delta T = \Delta\tau_r c/U$ is the real-time reversal motion lasts. The complete change of the wings in the flapping cycle is shown in Figure 3.

Numerical methodology: immersed boundary method

To simulate the flow past the hovering dragonfly with an inclined stroke in low Reynolds number, the incompressible Navier-Stokes equations are used:

$$\frac{\partial \boldsymbol{u}}{\partial t} + \boldsymbol{u}\cdot\nabla\boldsymbol{u} = -\nabla P + \frac{1}{Re}\nabla^2\boldsymbol{u} + \boldsymbol{f} \quad (2.1)$$

$$\nabla\cdot\boldsymbol{u} = 0 \quad (2.2)$$

$\boldsymbol{u}$ and p in the above equation is the velocity and pressure, and μ is the dynamic viscosity. $\boldsymbol{f}$ in Eq. (2.1) is the volume force exerted on fluid by the immersed boundary. In the direct force immersed boundary method, as shown in Figure 3, the fluid governing equation is solved on the background Eulerian grid, and the solid boundary is discretized into a series of LaGrange points at which the volume force is calculated and distributed back into the fluid grid. This process is also the key to the entire numerical simulation.

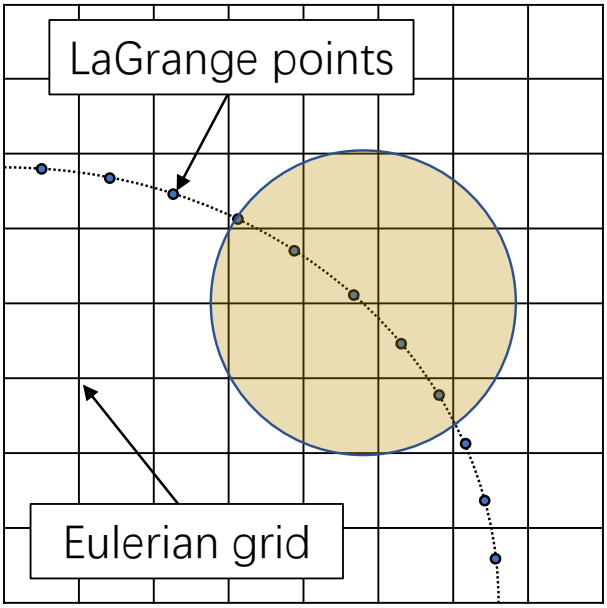


Figure 4. The schematics of the immersed boundary method

In our study, the immersed boundary method was proposed based on the predictor-corrector solver pisoFOAM. From the above Eq. (2.1), one can get:

$$\boldsymbol{f} = \frac{\partial \boldsymbol{u}}{\partial t} + \boldsymbol{u}\cdot\nabla\boldsymbol{u} + \nabla P - \frac{1}{Re}\nabla^2\boldsymbol{u} = \frac{\boldsymbol{u}^{n+1} - \boldsymbol{u}^n}{\Delta t} - \boldsymbol{rhs} \quad (2.3)$$

Where *n+1* and *n* represents different time step, and

$$\boldsymbol{rhs} = -\boldsymbol{u}\cdot\nabla\boldsymbol{u} - \nabla P + \frac{1}{Re}\nabla^2\boldsymbol{u} \tag{2.4}$$

In the predictor-corrector solver pisoFoam, the first step is to use the velocity and pressure at time *n* to calculate the predicted velocity $\mathbf{u}^*$:

$$\frac{\mathbf{u}^* - \mathbf{u}^{\mathrm{n}}}{\Delta \mathrm{t}} = \mathbf{rhs} \tag{2.5}$$

Considering the predicted velocity $\boldsymbol{u}^*$, the Eq. (2.3) is rewritten as

$$\boldsymbol{f} = \frac{\boldsymbol{u}^{n+1} - \boldsymbol{u}^*}{\Delta t} + \frac{\boldsymbol{u}^* - \boldsymbol{u}^n}{\Delta t} - \boldsymbol{rhs} = \frac{\boldsymbol{u}^{n+1} - \boldsymbol{u}^*}{\Delta t} \tag{2.6}$$

For the Lagrange points $\boldsymbol{X}_s$ at the solid boundary, the force term $\boldsymbol{F}_s$ is given by:

$$\boldsymbol{F}_s = \frac{\boldsymbol{U}_s^{n+1} - \boldsymbol{U}_s^*}{\Delta t} \tag{2.7}$$

Where $\boldsymbol{U}_s^{n+1}$ is the target velocity on sth LaGrange marker, $\boldsymbol{U}_s^*$ is evaluated through an interpolation from its Eulerian counterpart $\boldsymbol{u}^*$:

$$\boldsymbol{U}_s^* = \sum_{j\in G_h} \boldsymbol{u}_j^* \delta_h(\boldsymbol{x}_j - \boldsymbol{X}_s)\, h^3 \tag{2.8}$$

Here G_h denotes the collection of the Eulerian grid points, $\boldsymbol{x}_j$ is the coordinate of the jth Eulerian grid points, h is the Eulerian grid scale, and δ_h is discrete delta function which is applied to the information interaction between the Eulerian grid and LaGrange points:

$$\delta_h(\boldsymbol{x}_j - \boldsymbol{X}_s) = \frac{1}{h^3}\phi_h\left(\frac{x_j - X_s}{h}\right)\phi_h\left(\frac{y_j - Y_s}{h}\right)\phi_h\left(\frac{z_j - Z_s}{h}\right) \tag{2.9}$$

A 4-point piecewise function proposed by Peskin[15] is shown here:

$$\phi(r) = \begin{cases} \frac{1}{8}\left(3 - 2|r| + \sqrt{1 + 4|r| - 4r^2}\right) & 0 \le |r| < 1 \\ \frac{1}{8}\left(5 - 2|r| + \sqrt{-7 + 12|r| - 4r^2}\right) & 1 \le |r| < 2 \\ 0 & 2 \le |r| \end{cases} \tag{2.10}$$

The discrete delta function plays an important role in the immersed boundary method and directly determines the numerical accuracy. In the numerical simulation of dealing with moving objects, the common discrete function usually introduces non-physical numerical oscillation in the volume force. To solve this problem, Yang[16] proposed a smoothing technique to reconstruct a new discrete function from the conventional discrete function.

$$\phi'(r) = \int_{r-0.5}^{r+0.5} \phi(r')dr' \tag{2.11}$$

The following smoothed 4-point piecewise function is obtained by the smooth technique:

$$\phi'(r) = \begin{cases} \frac{3}{8} + \frac{\pi}{32} - \frac{r^2}{4} & |r| \le 0.5 \\ \frac{1}{4} + \frac{1-|r|}{8}\sqrt{-2 + 8|r| - 4r^2} - \frac{1}{8}\sin^{-1}\left(\sqrt{2}(|r| - 1)\right) & 0.5 \le |r| \le 1.5 \\ \frac{17}{16} - \frac{\pi}{64} - \frac{3|r|}{4} + \frac{r^2}{8} + \frac{|r| - 2}{16}\sqrt{-14 + 16|r| - 4r^2} + \frac{1}{16}\sin^{-1}\left(\sqrt{2}(|r| - 2)\right) & 1.5 \le |r| \le 2.5 \\ 0 & |r| \ge 2.5 \end{cases} \tag{2.12}$$

Then the volume force $\boldsymbol{F}_s$ at the Lagrange points need to be spread over the surrounding fluid Euler grid nodes:

$$\boldsymbol{f}_j = \sum_{s\in D_s} \boldsymbol{F}_s \delta_h(\boldsymbol{x}_j - \boldsymbol{X}_s)\cdot \Delta V_k \tag{2.13}$$

Where D_s is the collection of the Lagrange points, ΔV_k is the discrete volume for each Lagrange point[17]. In the correction step, the new velocity is obtained by solving the ns equation considering the volume force:

$$\frac{\boldsymbol{u}^{*,1}-\boldsymbol{u}^{n}}{\Delta t}=-\boldsymbol{u}^{*,1}\cdot\nabla\boldsymbol{u}^{*,1}-\nabla p+\frac{1}{Re}\nabla^{2}\boldsymbol{u}^{*,1}+\boldsymbol{f} \tag{2.14}$$

Then the pressure Poisson equation iteratively is solved from m=1 to m=N_s:

$do\ m=1{:}\,N_s$

$$\nabla^{2}p^{*,m}=-\nabla\cdot(\boldsymbol{u}^{*,m}\nabla\boldsymbol{u}^{*,m})+\nabla\cdot\boldsymbol{f} \tag{2.15}$$

$$\boldsymbol{u}^{*,m+1}=\{a_{ii}^{-1}\}(-\nabla p^{*,m}+\boldsymbol{f}+\{H\}) \tag{2.16}$$

$enddo$

a_{ii} and $\{H\}$ is the coefficient matrix of diagonal part and extra-diagonal part which is described in detail in Constant's study[18].

The original intention of the direct force method is to make the velocity of Lagrange points meet the non-slip boundary condition through the application of volume force. However, due to the errors generated in the process of velocity interpolation and volume force distribution, the points on the surface of objects do not meet the non-slip boundary condition. Multi-direct forcing scheme[19] can improve this situation by iteratively correction method, the specific calculation process is as follows:

$$\frac{\boldsymbol{u}^{*,0}-\boldsymbol{u}^{n}}{\Delta t}=-\boldsymbol{u}^{*,0}\cdot\nabla\boldsymbol{u}^{*,0}-\nabla p+\frac{1}{Re}\nabla^{2}\boldsymbol{u}^{*,0} \tag{2.17}$$

$do\ a=1{:}\,N_a$

$$\boldsymbol{U}_{s}^{*,a-1}=\sum_{j\in G_h}\boldsymbol{u}_{j}^{*,a-1}\delta_h\big(\boldsymbol{x}_j-\boldsymbol{X}_s\big)\,h^{3} \tag{2.18}$$

$$\boldsymbol{F}_{s}^{*,\,a-1}=\frac{\boldsymbol{U}_{s}^{d}-\boldsymbol{U}_{s}^{*,a-1}}{\Delta t} \tag{2.19}$$

$$\boldsymbol{f}_{j}^{*,a}+=\sum_{s\in D_s}\boldsymbol{F}_{s}^{*,\,a-1}\delta_h\big(\boldsymbol{x}_j-\boldsymbol{X}_s\big)\cdot\Delta V_k \tag{2.20}$$

$$\frac{\boldsymbol{u}^{*,a}-\boldsymbol{u}^{n}}{\Delta t}=-\boldsymbol{u}^{*,a}\cdot\nabla\boldsymbol{u}^{*,a}-\nabla p+\frac{1}{Re}\nabla^{2}\boldsymbol{u}^{*,a}+\boldsymbol{f}^{*,a} \tag{2.21}$$

$enddo$

$$\nabla^{2}p=-\nabla\cdot(\boldsymbol{u}^{*,Na}\nabla\boldsymbol{u}^{*,Na})+\nabla\cdot\boldsymbol{f}^{*,Na} \tag{2.22}$$

3. **Validation of the numerical method**

In this section, the accuracy of the immersed boundary method is verified by several classical fluid-structure coupling problems.

3.1 Flow past a stationary circular cylinder

Flow past a stationary circular cylinder is a typical example of testing numerical simulation methods in CFD. For the simulation of flow past a stationary cylinder, the computational domain and boundary conditions are shown in Figure 5. Then a domain size of $-15\mathrm{D}\le \mathrm{x}\le 25\mathrm{D}$ and $-15\mathrm{D}\le \mathrm{y}\le 15\mathrm{D}$ is used, $\mathrm{D}=1$ is the dimensionless diameter of the cylinder. The entrance on the left is the velocity inlet boundary condition with the dimensionless inlet velocity U, the right boundary is given a pressure outlet boundary condition. Other boundaries are imposed by Norman boundary conditions. The Reynolds number Re is defined as $\mathrm{Re}=\rho\mathrm{UD}/\mu$, The Strouhal number, drag and lift coefficients are defined by:

$$C_L=\frac{2F_L}{\rho U^{2}\mathrm{D}},C_D=\frac{2F_D}{\rho U^{2}\mathrm{D}},\mathrm{St}=\frac{\mathrm{D}f}{U} \tag{3.1}$$

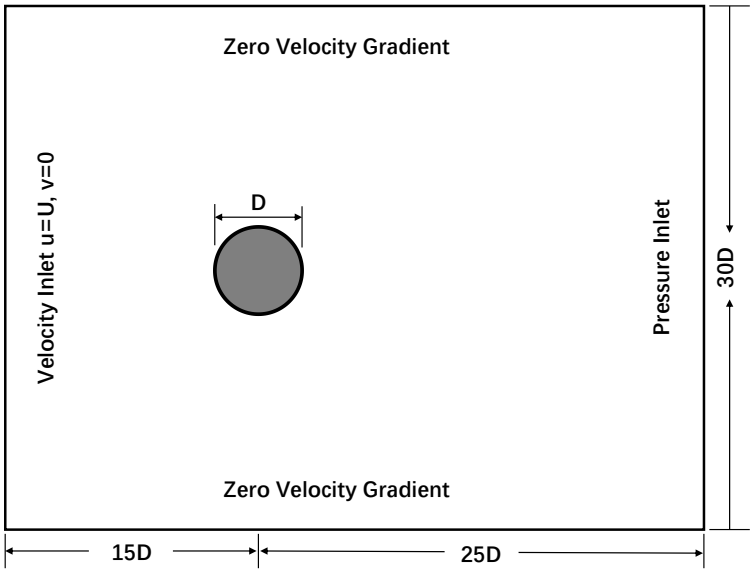


Figure 5. The computational domain, boundary conditions of flow past a stationary circular cylinder

Here F_L and F_D represent the instantaneous drag and lift, and f is the vortex shedding frequency. The simulations are carried out for the steady and unsteady flow regimes at two different Re = 30 and 100. For the steady flow at Re = 30, the most important parameters related to the wake, such as the length of the wake region I, the distance between the wake vortex center and the cylinder a, the distance between the wake vortexes b, the angle of separation measured from the x-axis θ and the drag coefficients C_D are compared with the results of previous experiments and numerical simulations in Table 1. To show that the multi-direct forcing scheme can make the boundary better satisfy the non-slip boundary condition, Figure 6 shows the circumferential distribution of the velocities interpolated to the Lagrange points on the cylindrical boundary. Note that the interpolated velocity obtained by the immersed boundary method with a multi-direct forcing scheme is close to the expected velocity (zero for a stationary cylinder). The instantaneous streamline and the vorticity contours were also plotted in Figure 7. A steady recirculating region is observed behind the cylinder.

Table 1. Comparison of wake parameters and drag coefficients for Re = 30

	I/D	a/D	b/D	θ^0	C_D
Present	1.69	0.57	0.54	49.15	1.79
Coutanceau and Bouard[20]	1.55	0.54	0.54	50.00	1.74
Pinelli and Naqavi[21]	1.70	0.56	0.52	48.05	1.80

For the unsteady flow at Re = 100, the periodic shedding of Karman vortex street in the wake region can be observed. In Table 2, the Strouhal number, drag coefficients, and lift coefficients obtained from the present solver are compared with that of the previous research. The flow field structure and corresponding time history curve of the lift resistance coefficient are shown in Figure 8 and Figure 9 respectively. These results are all in good agreement with previous studies.

Table 2. Comparison of parameters for Re = 100

	C_D	C_L	St
Present	1.43	0.34	0.167
Lai and Peskin[22]	1.45	0.33	0.165
Shin and Huang[23]	1.37	0.34	0.163

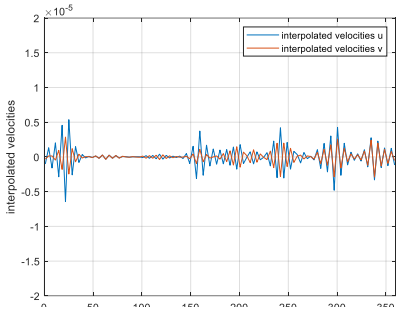


Figure 6. velocity along the cylinder

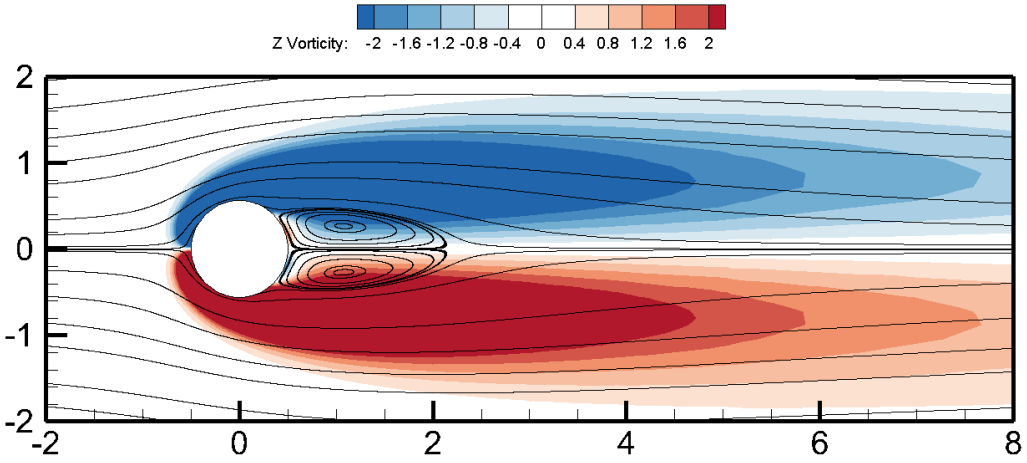


Figure 7. streamline and the vorticity contours at Re = 30

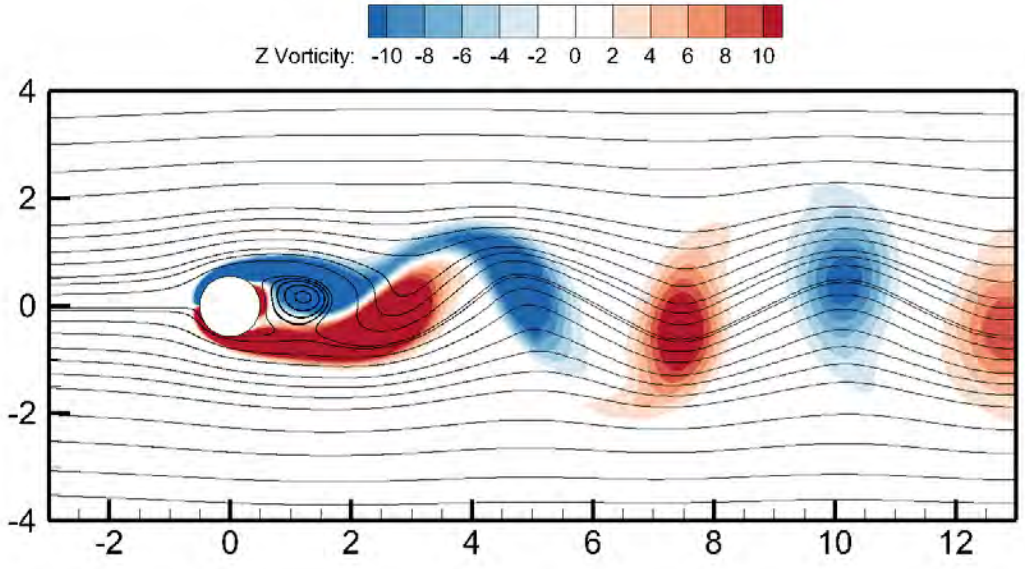


Figure 8. streamline and the vorticity contours at Re = 100

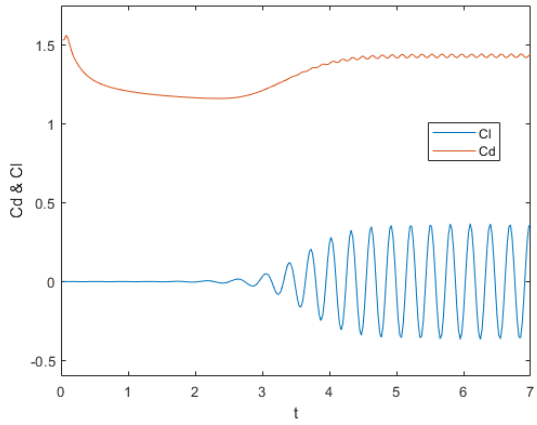


Figure 9. The curve of lift coefficient and drag coefficient over time at Re = 100

3.2 Flow past oscillating circular cylinder

To verify the ability of numerical methods to simulate the flow past moving objects, the flow past a cross-flow oscillating circular cylinder that undergoes a prescribed motion in the transverse direction using Eq. (3.2). is studied.:

$$y = -A\sin 2\pi f_e t \tag{3.2}$$

The calculations for Re = 185, $A/D = 0.2$ and $0.8 \leq f_e/f_0 \leq 1.2$ are performed, where f_0 is the natural shedding frequency from the stationary cylinder.

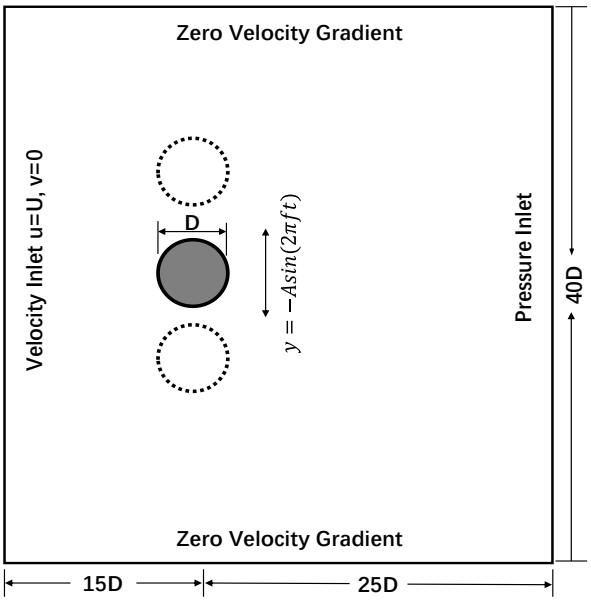


Figure 10. The computational domain, boundary conditions of flow past an oscillating circular cylinder.

The computational domain and boundary conditions are shown in Figure 10. Then a domain size of $-15\mathrm{D} \leq \mathrm{x} \leq 25\mathrm{D}$ and $-20\mathrm{D} \leq \mathrm{y} \leq 20\mathrm{D}$ is used, six frequency ratios were selected for numerical simulation, and the time history curve of the lift and drag coefficient was shown in Figure 11. The results agree well with those of others[24].

The purpose of this section is to verify the ability of the numerical method to handle the numerical simulation of flow past moving objects, and to verify the ability of the smoothing technique to suppress non-physical numerical oscillation. However, the numerical oscillation

of the volume force is not obvious in the case of the oscillating cylinder, so the verification work will be further described in the following paper.

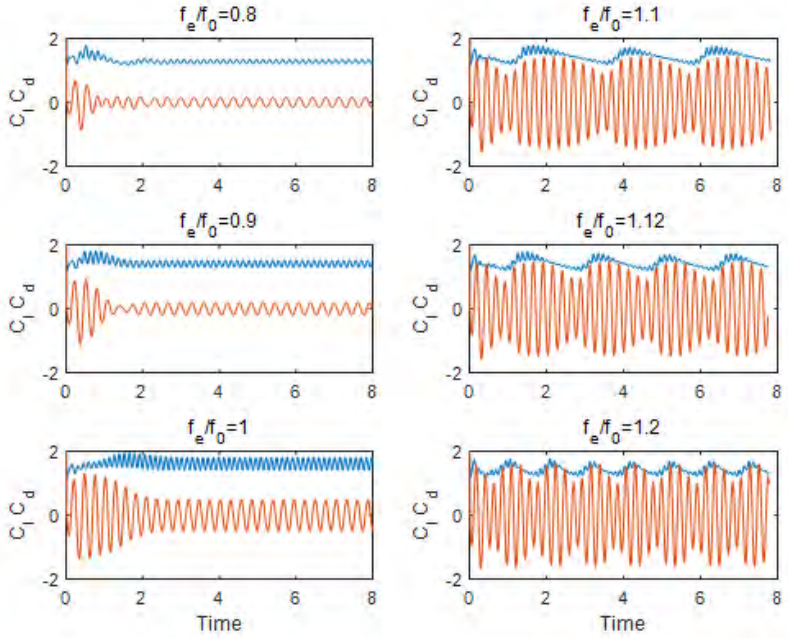


Figure 11. The curve of lift coefficient and drag coefficient over time for different f_e/f_0

The above results show that the fluid solver can well simulate the flow around a stationary cylinder at different Reynolds numbers and the flow around a moving cylinder at different oscillating frequencies. Compared with the conventional submerged boundary method, the introduction of the multi-direct forcing scheme makes the boundary better satisfy the non-slip boundary condition

4. Results and discussion

In this section, the flow over single-wing and tandem-wing of a hovering dragonfly is simulated, to save calculation time, only the forewing and hindwing of the unilateral side were simulated. Symmetric boundary conditions were applied to simulate the effect on the other side tandem wing. The size of the calculation domain was set to $15c * 15c * 10c$. The Cartesian grid is used for spatial dispersion of the fluid domain and consists of two subregions: the global cluster, which has large, uniform mesh spacing and the locality refine region, which is refined layer by layer around the immersed boundary.

Before exploring the aerodynamic characteristics of flapping wing dragonflies, the numerical method is further verified. We reproduced Sun's research work, selected the airfoil profile and kinematic parameters in his paper for numerical simulation of a single forewing, and calculated the aerodynamic force to compare with his results. As shown in Figure 12, the calculated lift and drag forces and their variation trends over time are quite consistent with the results obtained by the overset grids. Furthermore, the calculation model uses three different grid division strategies to verify the grid independence. From Figure 13 there is almost no difference between the force coefficients calculated by the three grid systems. In the subsequent work, we selected grid size 0.025c and time step size 1×10^{-6}s, that is, 2700 steps were calculated in one cycle. Figure 14 shows the comparison of the calculation results of two kinds of immersed boundary methods using the ordinary interpolation function and the smoothed interpolation function. It can be seen that the smoothing technique can effectively suppress the numerical oscillation of the aerodynamic coefficient

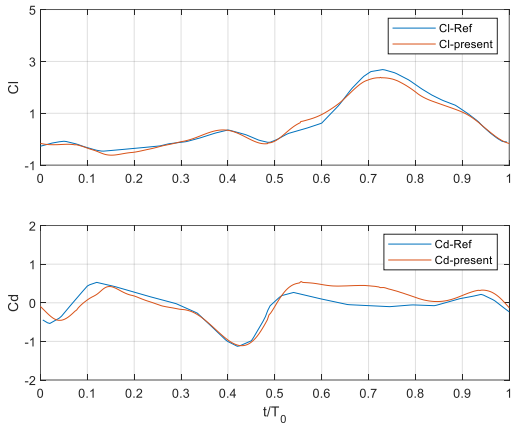


Figure 12. Comparison of the single forewing aerodynamic characteristics with references

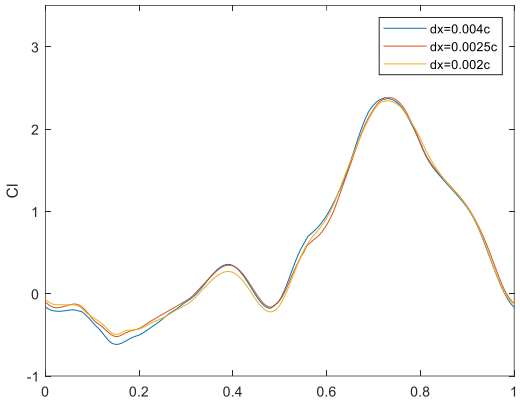


Figure 13. The curve of lift coefficient over time for different mesh size

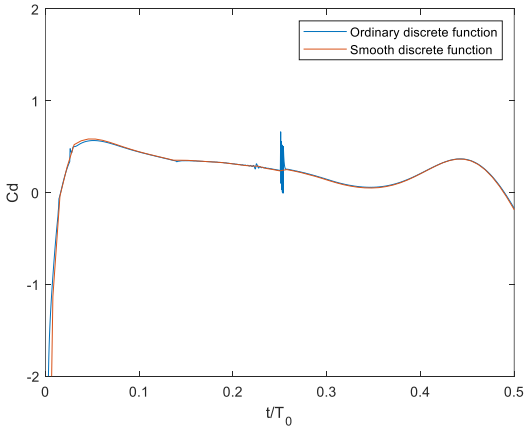


Figure 14. The curve of drag coefficient over time for different discrete function

4.1 Aerodynamic characteristics and flow structure of single-wing

In the calculation of this paper, the wing begins to flap in still air until the aerodynamic and flow structure exhibits favorable cyclic attributes and the calculation ends (approximately 4-5 cycles after the calculation begins). In the hover state, lift is the vertical component while thrust is the horizontal component of the total force, and the lift and thrust coefficients are defined as:

$$C_L = \frac{2F_L}{\rho U^2\left(S_f + S_h\right)}, C_T = \frac{2F_T}{\rho U^2\left(S_f + S_h\right)} \tag{4.1}$$

The aerodynamic power imparted to the air by each wing is calculated as the inner product of the velocity and the aerodynamic force[25]:

$$P = \sum_{i}^{N} (\boldsymbol{F_i} \cdot \boldsymbol{v_i}) \tag{4.2}$$

Figure 15 gives the lift and thrust coefficients of the fore- and hindwings. It can be seen that both the forewing and the hindwing produce a positive lift in the downstroke, and a negative lift in the upstroke. And the lift always reaches its peak in the middle of the half-stroke. Compared to the positive lift peak, the negative lift peak has a substantially smaller magnitude, so the average lift over some time is positive to overcome its gravity. In addition, the wing begins the pitch motion at the end of the upstroke, lift appears to have a smaller positive peak. Due to the larger wing area, the hindwing has a larger peak and average lift than forewing ($\overline{C_{L,sf}} = 0.54, \overline{C_{L,sh}} = 0.73$). Compared with lift, drag does not change dramatically throughout the cycle and the peak value exists only when the wing is in pitch motion.

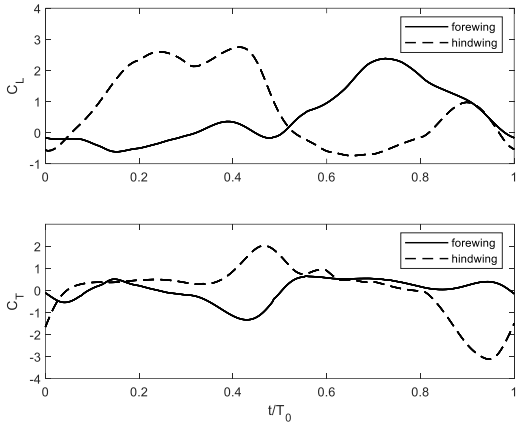


Figure 15. The curve of lift and drag coefficient over time for single-wing

In fluid-structure interaction problems, the force on the structure is directly related to the surrounding vortex structure and pressure distribution. Therefore, the vortex and pressure structure at various times during one cycle of forewing are presented in Figure 16 and Figure 17. Since the wing surface does not coincide with the fluid mesh, the pressure distribution of the wing surface is obtained by interpolation. To establish a correlation between the forces and the flow dynamics, we adopt a dimensionless parameter $\hat{t}$ to encapsulate the temporal progression within a stroke cycle. This parameter is defined such that $\hat{t} = 0$ marks the inception of the cycle, while $\hat{t} = 1$ signifies its culmination. The forewing starts its upstroke when $\hat{t} = 0$. At this time, the flapping velocity of the wing is zero and the motion state is dominated by the rotating motion. The vortex system structure formed in the downstroke of the last cycle and attached to the upper surface of the wing continues to shed and maintains the downward movement trend. Because the wing tip is farther from the wing root, the flapping velocity of the wing tip changes more widely, and the wing tip vortices are easier to separate. There is a positive pressure area on the upper wing tip due to the shedding of the wingtip vortex in advance, while the incomplete separation of the leading edge vortex makes the middle of the leading edge on the upper wing surface still covered by the negative pressure area. Then, while continuing to reduce the angle between the wing and the stroke plane, the flapping translation begins. When $\hat{t} = 0.125$, it can be seen from Figure 16 that the downward vortex will hinder

the movement of the wing and generate negative lift. However, since the angle between the wing and the stroke plane is small at this time, the negative lift force is not large. The rapid upward flapping causes vortices near the tower surface of the wingtip and a negative pressure zone on the lower surface of the wingtip The pitch motion of the wing ends at $\hat{t} = 0.2$, and at the middle part of the upstroke, the motion of the wing is completely composed of flapping, and the intensity of the wingtip vortex is significantly enhanced and spreads to the tailing edge, these locations become covered with black negative pressure zones, which is also one of the sources of negative lift in the upstroke. As the wing continues upstroke, the pitch motion begins at $\hat{t} = 0.3$. The wingtip vortex and tailing vortex begin to shed. The rapid change in the angle of attack induces additional circulation around the wing, and there is an analogy between this rotational cycle and the Magnus effect, which increases lift when the stroke is reversed[26]. In addition, for the symmetrical rotation of the dragonfly, a positive peak and a negative peak of the forewing lift in Figure 15 are also consistent with Dickinson's conclusion. When $\hat{t} = 0.5$,

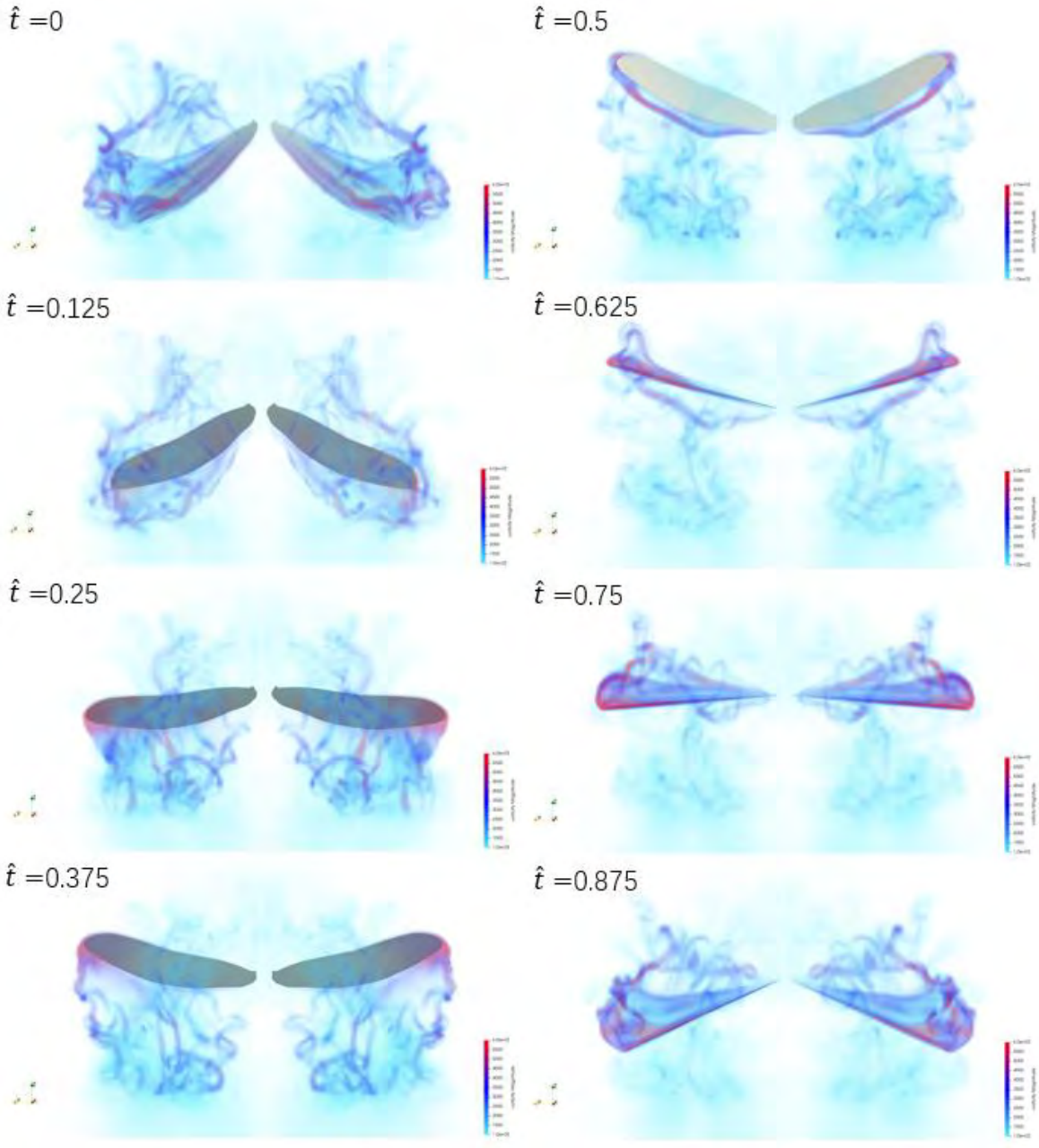


Figure 16. Vortex structure at various times during one cycle of forewing

velocity of the trailing edge since the axis of pitch motion is about a quarter chord away from the leading edge, the pitching is higher, so the trailing edge vortices are almost completely shed, while the wing tip vortices are still attached to the lower surface of the wing due to the low pitching velocity near the wing tip. The positive pressure zone caused by the pitch motion on the trailing edge of the upper surface can also explain the small negative lift peak mentioned earlier. As the wing continues to flap downward, it squeegees the gas below, causing the lower surface of the wing to be covered by positive pressure, which is stronger near the wing tip where the flapping velocity is faster. The shear layer detaches from the leading edge and reconnects to the top surface. This detachment and subsequent reattachment give rise to a large vortex known as the LEV[27]. This also causes the upper surface to be covered by a negative pressure zone. It also makes the wing maintain a high lift at a high angle of attack. This lift mechanism is called a delayed stall. In the figure, the vortex system structure slightly away from the wing is not generated by the falling off of the upper surface vortices but is evolved from the wing tip vortex not completely falling off at the end of the upper flap of the wing. At $\hat{t} = 0.75$, the angle between the wing chord and the horizontal plane is almost zero, the leading edge vortex strength increases and is still stably attached to the wing surface, and the positive ground pressure area of the lower surface is also enhanced when the flapping speed reaches the peak, and the pressure difference between the upper and lower surfaces of the wing causes the peak of the lift coefficient. The wing continues to flap downward, and the wing tip vortex starts to shed, forming a tailing vortex in the stroke plane. The transition from the negative pressure zone to the positive pressure zone on the upper surface of the wing tip for the shedding of the wingtip vortex and the strength of the positive pressure zone on the lower surface decrease with the reduced flapping velocity, which leads to the attenuation of the lift coefficient.

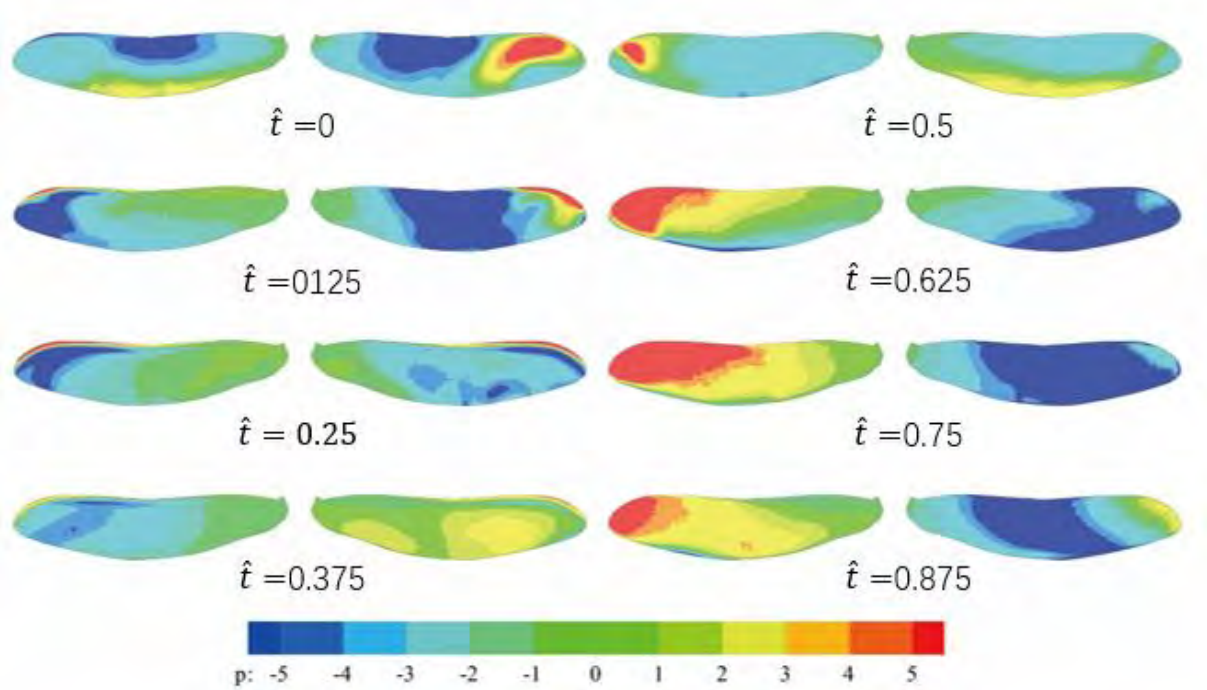


Figure 17. Pressure contour at various times during one cycle of forewing (the left side is the lower surface and the right side is the upper surface)

4.1 The effect of interaction between the fore- and hindwings

In order to investigate the interaction effect between fore- and hindwings on the aerodynamic characteristics of dragonflies, we calculate the flow of lateral tandem wing. The phase difference between the fore- and hindwings is set at 180°. The instantaneous lift and thrust

coefficient curves are plotted in Figure 18. It reveals a distinct pattern in the aerodynamic performance over a single cycle, characterized by two significant lift peaks. The first peak occurs during the initial phase, corresponding to the downstroke of the hindwing, while the second peak materializes in the latter part of the cycle, coinciding with the downstroke of the forewing. It is important to highlight that the presence of these two pronounced CL peaks, alternating between the cycle's first and second halves, can make the flight trajectory more stable and smoother. The total average lift coefficient of the wings is 1.12, corresponding to an average lift of 7.68mN, which is almost equal to the gravity of the dragonfly (7.54mN). The forewing and hindwing contribute to the total average vertical force by 44% and 56%, respectively. The vertical force exerted by the hindwing is 1.28 times that of the forewing. It should be noted that the area of the hindwing is 1.32 times larger than that of the forewing. Consequently, the relatively larger vertical force experienced by the hindwing is primarily due to its greater size. And the total average thrust coefficient of the wings is negligible. The average force coefficients and average power over the period for single/tandem wings are shown in Table 3. For both the fore- and hindwings, the vertical force coefficient on a single wing (i.e. without interaction) is a little larger than that with interaction. For the forewing, the interaction results in an 8% reduction in the average vertical force coefficient compared to that of a single wing. The reduction for the hindwing is 12% relative to that of a single wing. The impact of this interaction is relatively modest and does not favor the generation of vertical forces. The power expenditure is reduced by 8.2% and 10.6% for forewing and hindwing, respectively. The vortex and pressure structure at various times during one cycle of the forewing are presented in Figure 18 and Figure 19. At $\hat{t} = 0$, The forewing flaps upward from the lowest point while the hindwing flaps downward from the highest point. At this time, the distance between the fore- and hindwings is large, and the influence of tandem interactions is small. The slight difference

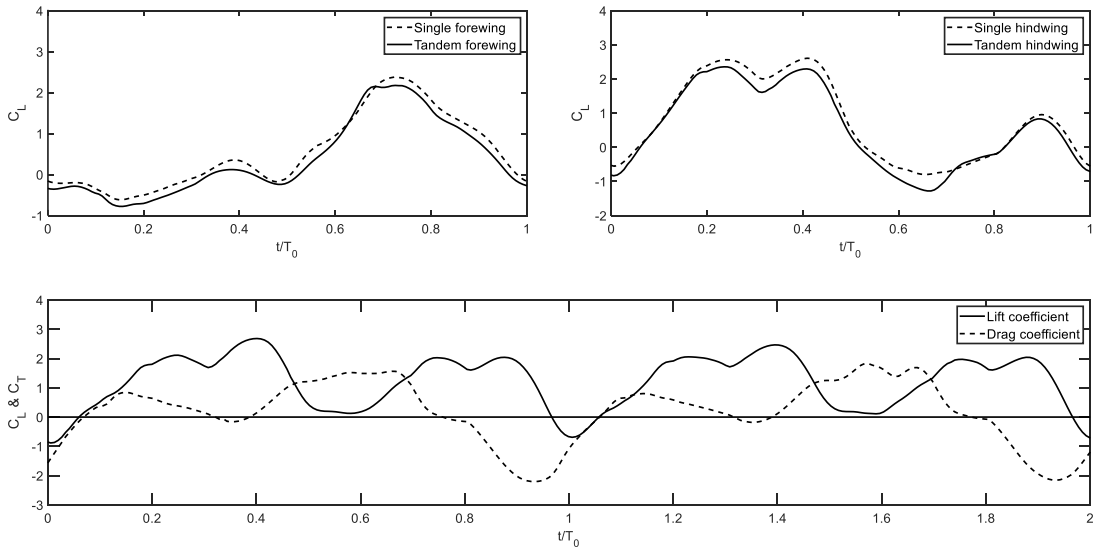


Figure 18. Force coefficient over time for single/tandem wing

Table 3. Comparison of force coefficient and average power for single/tandem wing

Wing	States	$\overline{C_L}$	$\overline{C_T}$	$\bar{P}(mW)$
Single wing	Forewing	0.54	0.019	7.09
	Hindwing	0.73	0.033	13.2
Tandem wing	Forewing	0.50	0.009	6.51
	Hindwing	0.64	0.013	11.8

in lift coefficient is due to the influence of a more complex vortex system structure generated in the last cycle by tandem wings on the flow field near the wings. At this time, the movement of the wing is dominated by the pitch movement. With the change of the wing stroke, the trailing edge vortex and wingtip vortex on the lower surface of the hindwing, the leading edge vortex, trailing edge vortex, and wing tip vortex attached to the upper surface of the forewing begin to shed. Different from the forewing, the hindwing has a larger span in the chord direction near the wing root. Therefore, during the pitch motion, the vortex system structure is induced at the wing root, and the upward velocity of the rear edge compresses the gas above the hindwing, forming an obvious positive pressure area on the upper surface of the wing, together with the positive pressure zone caused by the wingtip vortex attached to the upper surface of forewing leads to the negative lift of the dragonfly at this time. At $\hat{t} = 0.125$, the hindwing flapping down at a larger Angle of attack has got rid of the vortices generated during the last cycle, and the direction of the trailing edge motion change and stops squeezing the upper side gas of the hindwing. Due to the close distance between the left and right hindwings, vortices were still attached near the wing root. Meanwhile, the larger flapping speed at the wing tip caused vortices to begin to occur at the leading edge near the wing tip. Similarly, the front wing also produces a wingtip vortex. When the hindwing keeps flapping downwards, the vortex structure on the upper surface remains attached and its strength keeps escalating. The distance between the fore- and hindwing is the closest at $\hat{t} = 0.25$. At this time, the forewing, which flaps upwards, induces an upward velocity between the fore- and hindwing. Although this is insufficient to

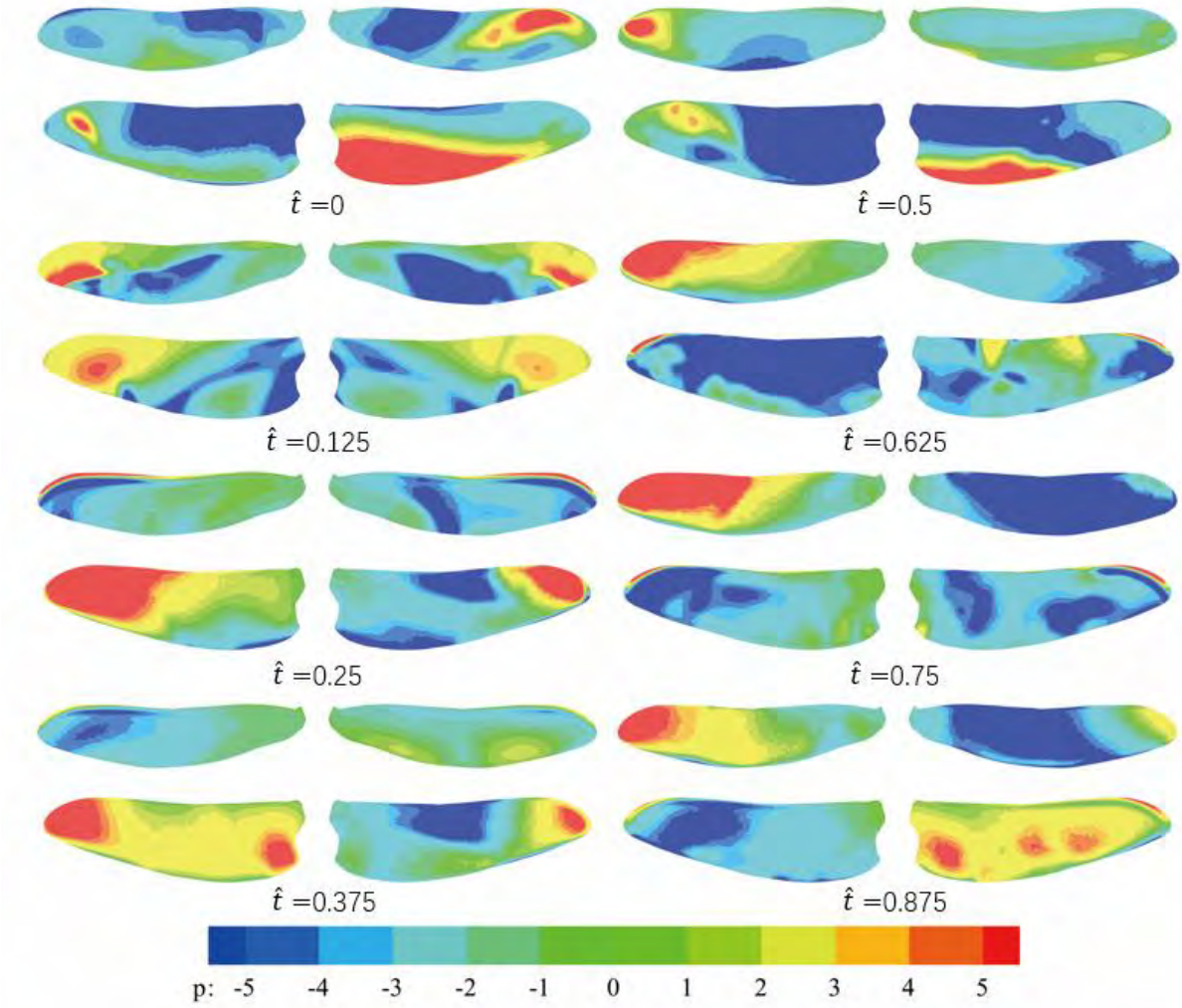


Figure 19. Pressure contour at various times during one cycle of tandem wing (the left side is the lower surface and the right side is the upper surface)

alter the direction of the gas flow near the leading edge of the hindwing, it influences the adhesion of the leading-edge vortex and leads it to detach prematurely. This effect is the most prominent at the wing tip because the wing tip of the forewing moves more rapidly, and a positive pressure zone emerges at the wing tip on the upper surface of the hindwing. However, compared with the single hindwing, it can also be seen from Figure 18 that the lift attenuation is not severe, and the dragonfly can still maintain a high lift. This indicates that the main mechanism for generating lift of the hovering dragonfly is the mutual squeeze between the flapping wing at a small angle of attack and the gas in the direction of motion. The hindwing continues to flap downward. The central position of the leading edge is less affected by the forewing, and the leading edge vortex is still attached. At moment $\hat{t} = 0.375$, this position is still covered by the negative pressure zone, and the positive pressure zone on the lower surface is still providing lift for the hindwing, and the vortices generated by the wing pitch movement at the end of the upper stroke are completely separated at this time. However, the forewing is not greatly affected in the whole upstroke process, neither the flow field structure nor

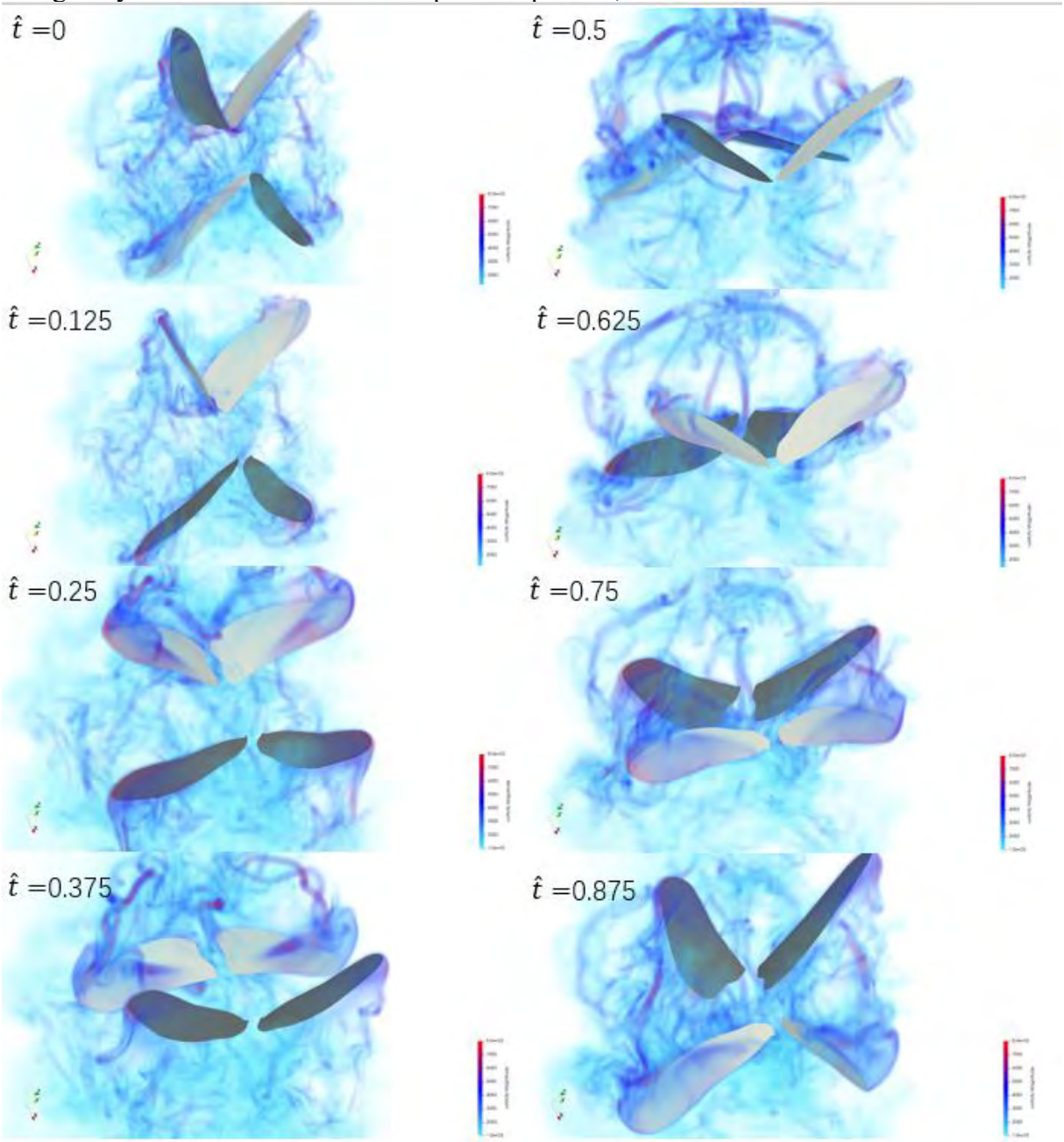


Figure 20. Vortex structure at various times during one cycle of tandem wing

the surface pressure distribution. At the same time, the wing body is almost parallel to the flapping plane, which makes the forewing suffer little aerodynamic force. After $\hat{t} = 0.375$, the motion modes of the fore- and hindwing are transformed from flapping to pitching, and the trailing edge vortex begins to appear. The pressure distribution of the trailing edge and wingtip of the lower surface of the forewing, and the trailing edge of the upper surface of the hindwing all change. The pitch of the hindwing allows its trailing edge to continue to have a greater downward velocity, and at $\hat{t} = 0.4$, the lift coefficient benefits from the larger wing area and even reaches a second peak. At $\hat{t} = 0.5$, except for the wingtip vortices on the front wing, the rest of the vortices began to fall off, there is a positive pressure area at the lower surface tip of the forewing, while the lower surface pressure of the hindwing rapidly changes from positive pressure to negative pressure due to the attenuation of the downward velocity. And then the stroke covert, as the forewing begins to flap downward, the wing tip vortices on the lower surface dissipate rapidly, and the leading edge vortices, trailing edge vortices, and wingtip vortex on the upper surface begin to appear. A negative pressure zone is formed on the upper surface of the forewing, and a positive pressure zone is formed on the lower surface due to extrusion. With the acceleration of the flapping speed of the forewing, the pressure difference between the upper and lower surfaces of the front wing further increases, resulting in a rapid increase in lift. Under the influence of the hindwing, the tailing edge vortex in the middle of the forewing appears to fall off, and the negative pressure area on the upper surface is slightly weakened, but it has little effect on the lift coefficient. At the end of the upstroke of the hindwing, the angle of attack brought about by the pitch movement changes rapidly, which weakens the negative pressure area of the lower surface of the wing, and the lift coefficient reaches a small peak. At the end of the forewing downstroke, the wing tip vortices appear to fall off. The positive pressure appeared at the wing tip of the upper surface, and the positive pressure area of the lower surface weakened due to the decrease in flapping speed, and the lift coefficient weakened accordingly.

Conclusions

In this paper, by combining the multi-forcing scheme and smoothing technique of the interpolation function, we developed the immersed boundary method based on PISO algorithm on OpenFOAM software. After verification, we found that it can complete the numerical simulation of incompressible complex flows well. The multi-forcing scheme makes the velocity near the solid boundary better meet the non-slip boundary condition. The smoothing technique of the interpolation function effectively avoids the numerical oscillation of the volume force caused by the processing of the dynamic boundary problem. At the same time, we use this solver to complete the numerical simulation of the single and tandem wings of the hovering dragonfly and find that the dragonfly can generate enough lift to overcome gravity, and the hindwing can generate higher lift due to its larger wing area. Combined with the pressure distribution of the wing surface and the vortex structure, the lift mechanism of the dragonfly was analyzed, including the stable leading edge vortex attachment during the downstroke, the delayed speed mechanism at a high angle of attack, the mutual extrusion between the wing and the surrounding fluid, and the rotation circulation at the end of the upstroke, etc. At the same time, the influence of tandem wing interaction on the aerodynamic characteristics of dragonflies was studied. It is found that tandem wing interaction is more obvious in the middle part of each stroke. The pressure distribution of the wing surface is changed by inducing the leading edge vortex and wingtip vortex of the hindwing and the trailing edge vortex of the forewing to fall off in advance, and then the lift force is changed. The tandem wing interaction causes the lift force of the fore- and hindwings to decrease by 8% and 12%, respectively.

References

[1] Ellington, C. P. (1984). The Aerodynamics of Hovering Insect Flight .1. The Quasi-Steady Analysis, *Philosophical Transactions of the Royal Society of London Series B-Biological Sciences* **305**(1122), 1-15.

[2] Birch, J. M. and Dickinson, M. H. (2003). The Influence of Wing–Wake Interactions on the Production of Aerodynamic Forces in Flapping Flight, *Journal of Experimental Biology* **206**(13), 2257-2272.

[3] Van Truong, T., Le, T. Q., Park, H. C. and Byun, D. (2017). Experimental and Numerical Studies of Beetle-Inspired Flapping Wing in Hovering Flight, *Bioinspiration & Biomimetics* **12**(3).

[4] Wang, Z. J. (2000). Two Dimensional Mechanism for Insect Hovering, *Physical Review Letters* **85**(10), 2216-2219.

[5] Gong, C., Han, J., Yuan, Z., Fang, Z. and Chen, G. (2019). Numerical Investigation of the Effects of Different Parameters on the Thrust Performance of Three Dimensional Flapping Wings, *Aerospace Science and Technology* **84**, 431-445.

[6] Kang, C. K., Aono, H., Cesnik, C. E. S. and Shyy, W. (2011). Effects of Flexibility on the Aerodynamic Performance of Flapping Wings, *Journal of Fluid Mechanics* **689**, 32-74.

[7] Lua, K. B., Zhang, X. H., Lim, T. T. and Yeo, K. S. (2015). Effects of Pitching Phase Angle and Amplitude on a Two-Dimensional Flapping Wing in Hovering Mode, *Experiments in Fluids* **56**(2).

[8] Peskin, C. S. (1972). Flow Patterns around Heart Valves - Numerical Method, *Journal of Computational Physics* **10**(2), 252-&.

[9] Mittal, R. and Iaccarino, G. (2005). Immersed Boundary Methods, *Annual Review of Fluid Mechanics* **37**(1), 239-261.

[10] Norberg, R. A. (1972). Pterostigma of Insect Wings an Inertial Regulator of Wing Pitch, *Journal of Comparative Physiology* **81**(1), 9-&.

[11] Sun, M. and Lan, S. L. (2004). A Computational Study of the Aerodynamic Forces and Power Requirements of Dragonfly (Aeschna Juncea) Hovering, *Journal of Experimental Biology* **207**(11), 1887-1901.

[12] Azuma, A. and Watanabe, T. (1988). Flight Performance of a Dragonfly, *Journal of Experimental Biology* **137**, 221-252.

[13] Peng, L., Zheng, M., Pan, T., Su, G. and Li, Q. (2021). Tandem-Wing Interactions on Aerodynamic Performance Inspired by Dragonfly Hovering, *Royal Society Open Science* **8**(8).

[14] Wang, J. K. and Sun, M. (2005). A Computational Study of the Aerodynamics and Forewing-Hindwing Interaction of a Model Dragonfly in Forward Flight, *Journal of Experimental Biology* **208**(19), 3785-3804.

[15] Peskin, C. S. (2003). The Immersed Boundary Method, *Acta Numerica* **11**, 479-517.

[16] Yang, X., Zhang, X., Li, Z. and He, G.-W. (2009). A Smoothing Technique for Discrete Delta Functions with Application to Immersed Boundary Method in Moving Boundary Simulations, *Journal of Computational Physics* **228**(20), 7821-7836.

[17] Uhlmann, M. (2005). An Immersed Boundary Method with Direct Forcing for the Simulation of Particulate Flows, *Journal of Computational Physics* **209**(2), 448-476.

[18] Constant, E., Favier, J., Meldi, M., Meliga, P. and Serre, E. (2017). An Immersed Boundary Method in Openfoam : Verification and Validation, *Computers & Fluids* **157**, 55-72.

[19] Wang, Z., Fan, J. and Luo, K. (2008). Combined Multi-Direct Forcing and Immersed Boundary Method for Simulating Flows with Moving Particles, *International Journal of Multiphase Flow* **34**(3), 283-302.

[20] Coutanceau, M. and Bouard, R. (2006). Experimental Determination of the Main Features of the Viscous Flow in the Wake of a Circular Cylinder in Uniform Translation. Part 1. Steady Flow, *Journal of Fluid Mechanics* **79**(2), 231-256.

[21] Pinelli, A., Naqavi, I. Z., Piomelli, U. and Favier, J. (2010). Immersed-Boundary Methods for General Finite-Difference and Finite-Volume Navier–Stokes Solvers, *Journal of Computational Physics* **229**(24), 9073-9091.

[22] Lai, M.-C. and Peskin, C. S. (2000). An Immersed Boundary Method with Formal Second-Order Accuracy and Reduced Numerical Viscosity, *Journal of Computational Physics* **160**(2), 705-719.

[23] Shin, S. J., Huang, W. X. and Sung, H. J. (2008). Assessment of Regularized Delta Functions and Feedback Forcing Schemes for an Immersed Boundary Method, *International Journal for Numerical Methods in Fluids* **58**(3), 263-286.

[24] Guilmineau, E. and Queutey, P. (2002). A Numerical Simulation of Vortex Shedding from an Oscillating Circular Cylinder, *Journal of Fluids and Structures* **16**(6), 773-794.

[25] Aegerter, C. M., Zheng, L., Hedrick, T. L. and Mittal, R. (2013). Time-Varying Wing-Twist Improves Aerodynamic Efficiency of Forward Flight in Butterflies, *PLoS ONE* **8**(1).

[26] Dickinson, M. H., Lehmann, F. O. and Sane, S. P. (1999). Wing Rotation and the Aerodynamic Basis of Insect Flight, *Science* **284**(5422), 1954-1960.

[27] De Manabendra, M., Sudhakar, Y., Gadde, S., Shanmugam, D. and Vengadesan, S. (2024). Bio-Inspired Flapping Wing Aerodynamics: A Review, *Journal of the Indian Institute of Science*.

Numerical study of inter-floating platform mooring systems based on the SPH method

†Weibo Du, *Peigang Jiao, Kangning Li, and Jiaming Ding
College of Mechanical Engineering, Shandong Jiaotong University, China.
*Presenting author: 2811343939@qq.com
†Corresponding author: jiaopeigang@163.com

Abstract

Based on the smoothed particle hydrodynamics (SPH) method, the numerical study of mooring mode between floating platforms is carried out in this paper. By analyzing the performance of floating system mooring system, the stability and safety of offshore platform are improved. By using the SPH method, the interaction and dynamic response of the simulated floating platform group under ocean factors such as waves are studied. As a particle-based simulation technique, SPH method can effectively deal with free surface flow, multiphase flow and their interaction problems, and is especially suitable for simulating complex hydrodynamic phenomena related to the dynamic response of the floating platform. In this paper, two mooring system models between floating platforms are constructed, and each floating platform is connected to each other by a predetermined cable method. By setting different mooring modes between the floating platform groups, the influence of different mooring configurations on the overall system stability in the actual marine environment is simulated. The research table shows that the reasonable mooring system setting between the floating platform groups can significantly enhance the wind and wave resistance and structural safety of the floating platform group system.

Keywords: SPH, Mooring System, Numerical Simulation, Fluid-Structure Interaction

Introduction

Compared with onshore wind power generation, offshore wind power generation not only has advantages in resource occupation, noise and visual pollution, but also has larger installed capacity and sufficient wind energy resources. It has become a new direction of wind power development and has huge untapped potential. In recent years, in order to meet the needs of offshore platform development, the research of offshore mooring theory and technology has been booming. The engineering practice shows that compared with the fixed foundation wind power generation equipment, the floating offshore wind turbine is in a more complex environment. One of the key technologies for the stability of the floating platform is the stability of its mooring system. With the increase of water depth, the load of the mooring line increases rapidly, and the instability of the floating platform motion increases. On the basis of ensuring the stability of the unit 's power generation, the dynamic load of the unit is reduced to ensure the floating offshore wind turbine. Reliable operation in harsh operating

environments. Therefore, it is of great significance to study the mooring system between floating offshore wind turbines for the further promotion of offshore wind power generation.

Method application

SPH (Smoothed Particle Hydrodynamics) method [1] is a particle-based numerical method. It was originally used for the evolution calculation of celestial bodies in the field of astrophysics, and later gradually applied to fluid mechanics, physics, materials science and other fields. The SPH method was first proposed by Lucy and Gingold [2-3] in the field of astrophysics. The main idea is to divide the fluid continuum into a large number of discrete particles, and smooth it by kernel function, so as to obtain the physical quantity of the fluid. In this paper, by establishing different mooring systems between floating offshore wind turbines, the dynamic response of floating platform groups in the marine environment is simulated, which provides theoretical support for the optimization of mooring system structure between floating platform groups.

Coupling of SPH with MoorDyn

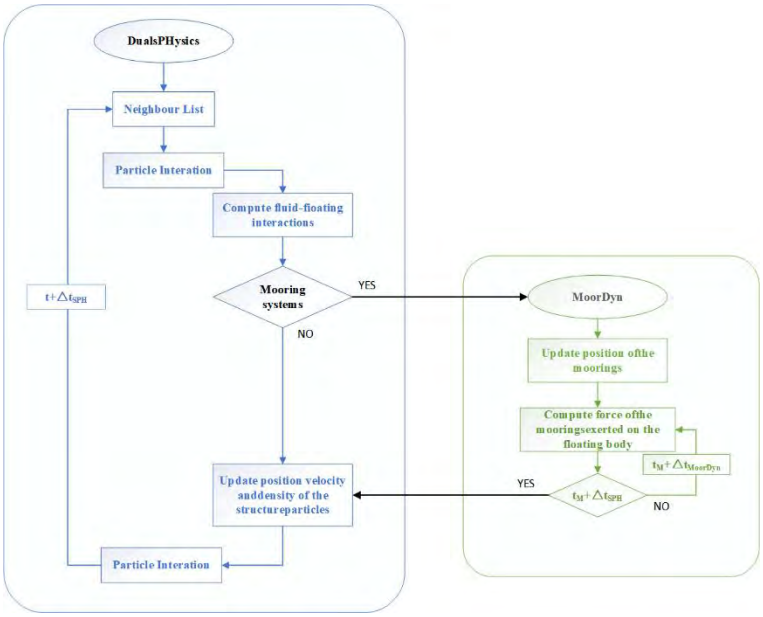


Figure 1. Flowchart of the Dualsphysis-Moordyn coupling

The coupling of DualSPHysics and MoorDyn can realize the simulation of the interaction between the offshore platform and the ocean wave. The force of the wave on the platform is simulated by DualSPHysics, and then the motion and rotation (V, Ω, R_0) of the solution are passed to MoorDyn and used as the input of the mooring line kinematics to update the motion state of the platform. Then MoorDyn simulates the motion state of the mooring line motion in the time step, and calculates the force (dV/dt, $d\Omega/dt$) at the connection of the guide cable. These forces are transmitted back to DualSPHysics. The final resultant force acting on the floating structure is obtained to calculate the final motion and rotation of the platform structure.

SPH approximation equations

SPH method is a Lagrangian particle method for simulating fluid flow. Different from grid methods such as finite element method and finite difference method, it does not need to divide the computational domain into grids. Instead, it discretizes the computational domain into a series of interacting particles for the simulation of fluid dynamics. The discrete Navier-Stokes equation performs local integration at each position of these particles according to the physical properties of the surrounding particles. The particles carry various physical quantities, including mass, density, velocity, acceleration, energy, etc., and interact with each other through kernel functions. Adjacent particles are determined by distance-based functions. The function can be two-dimensional or three-dimensional, and the relevant feature length or smooth length is usually expressed as h. In each time step, new physical quantities are calculated for each particle, and then they move according to the updated values [4].

SPH method is a kind of integral interpolation method, using the integral equation based on interpolation function, the conservation law of continuous hydrodynamics is transformed from their partial differential equation form into a suitable particle-based simulation form. This interpolation or weighting function is called kernel function. The basic principle is to approximate any function $F(r)$ by integral interpolation:

$$F(r) = \int F(r')W(r - r', h)dr' \tag{1}$$

The particle estimation of the field function at particle *i* is finally:

$$F(r_i) \equiv \sum_j F(r_j)\frac{m_j}{\rho_j}W_{ij} \tag{2}$$

Among them.

$$W_{ij} = W(r_i - r_j, h) \tag{3}$$

Among them, ρ_j is the density of particle j (j = 1, 2, · · ·, N), where *N* is the total number of particles in the support domain of particle *j*, and m_j is the mass of particle *j*, so that the particle volume is $V_j = m_j / \rho_j$, $W(r_i - r_j, h)$ is a smooth function. The smooth kernel must satisfy the conditions of normalization, nonnegativity, decay, compactness, Dirac function, symmetry and smoothness [5].

In the SPH method, the choice of kernel function has a great influence on the accuracy and computational efficiency of the simulation results. In this paper, the Wendland kernel function [6] is adopted. The Wendland function has better smoothness and differentiability, and it is better when dealing with the problems of high fluid viscosity and large change of liquid surface morphology :

$$W(r,h)=\alpha_D\left(1-\frac{q}{2}\right)^4(2q+1)\quad 0\le q\le 2 \tag{4}$$

In the formula : where α_D is applied to 2D take $7/4\pi h^2$, 3D take $21/\pi h^3$, q is the relative length, $q=r/h$. In this paper, only the kernel with an influence domain of 2h is considered.

Control equations

Each floating platform is set as a rigid body [7]. According to the applied kernel function and the smooth length, the sum of the contributions of all the surrounding fluid particles to the boundary particles is taken as the net force on the boundary particles. Therefore, each boundary particle k is subjected to a force, and each unit mass is shown in Equation (5):

$$f_k=\sum_{a\in WP_s} f_{ka} \tag{5}$$

Where f_{ka} is the unit mass force exerted by the fluid particle a on the boundary particle *k*, as shown in equation (6):

$$m_k f_{ka}=-m_a f_{ak} \tag{6}$$

For the motion of the floating platform, the basic equations of rigid body dynamics can be used:

$$M\frac{dV}{dt}=\sum_{k\in BPs} m_k f_k \tag{7}$$

$$I\frac{d\Omega}{dt}=\sum_{k\in BP_s} m_k\left(r_k-R_0\right)\times f_k \tag{8}$$

In the formula, M is the mass of the floating platform, I is the moment of inertia, V is the speed, Ω is the rotational speed, and R_0 is the center of mass. Formulas (7) and (8) are integrated in time to predict the values of V and Ω at the beginning of the next time step. Each boundary particle in the floating platform has a velocity, as shown in Formula (9):

$$u_k=V+\Omega\times\left(r_k-R_0\right) \tag{9}$$

Finally, the boundary particles in the floating platform are moved by time integration of formula (9).

In the SPH form, the fluid is regarded as a weakly compressible fluid, and the fluid pressure is determined by the state equation:

$$P = B\left[\left(\frac{\rho}{\rho_0}\right)^{\gamma} - 1\right] \tag{10}$$

Among them, $\gamma = 7$, $B = c_0^2 \rho_0 / \gamma$, $\rho_0 = 1000 kg/m^3$ are the reference densities, and $c_0 = c(\rho_0) = \sqrt{\partial P / \partial \rho}$ is the sound velocity at the reference density [8].

Boundary condition

The boundary condition used in this paper is the dynamic boundary condition (DBC) [9]. When the fluid particles are close to the boundary, the distance between the particles and the fluid particles is less than twice the smooth length (h), the density of the affected boundary particles increases, resulting in an increase in pressure. In turn, this causes the repulsive force to be applied to the fluid particles due to the pressure term in the momentum equation. The static floating platform is only represented by the fixed particles with pressure in the equation of state. The boundary is easy to set, and the calculation is relatively stable and efficient, so that the floating platform provides reliable numerical simulation.

Numerical model

The mooring system of the floating offshore wind turbine is mainly composed of several sets of cable systems. Some of these cable systems are distributed around the platform and connect the floating platform with the seabed, and the other part is distributed between the floating platforms. The numerical model of this experiment is shown in Figure 2. The left side of the tank is equipped with a wave maker, and the right side is equipped with a wave absorbing area [10].

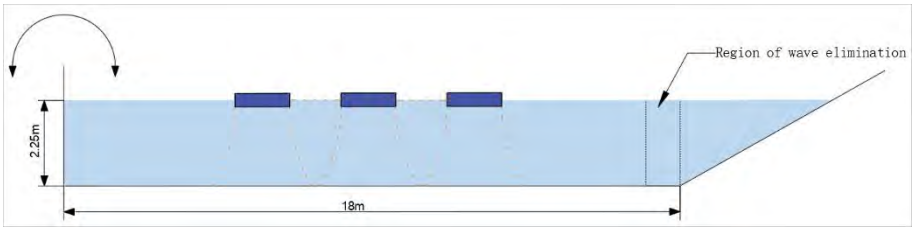


Figure 2 Schematic diagram of the numerical tank

The mooring system of the floating platform used in this paper is composed of multiple ' Lines ' mooring line units. These units are constructed by segmentation method. A cable is divided into 20 lines of equal size and connected to 21 nodes. The upper end of the mooring line is connected to the base endpoint of the floating platform, and the lower end is fixed on the seabed, and it is in a catenary state under a certain pre-tension, as shown in Figure 3 [11].

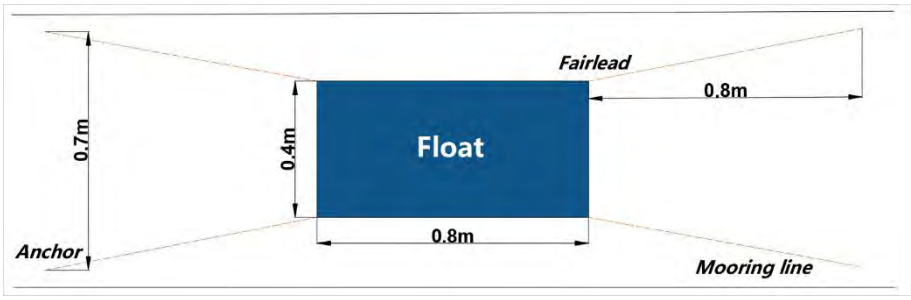
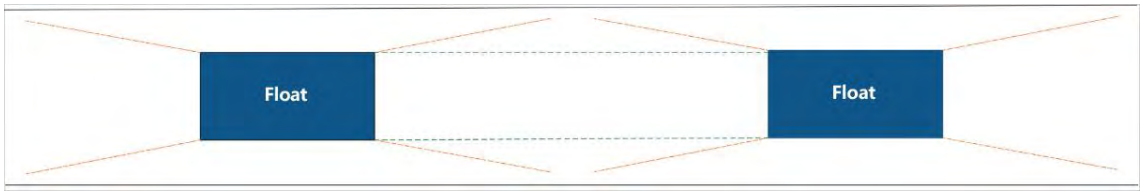
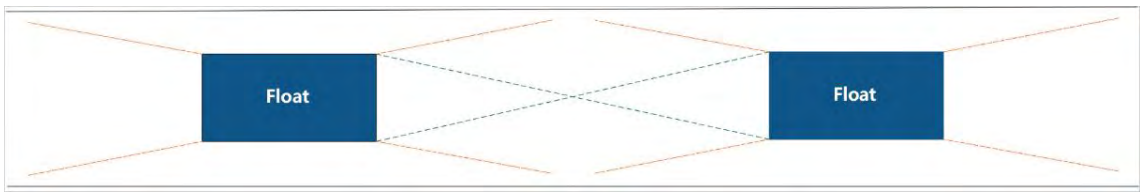


Figure 3 Schematic diagram of a single floating platform mooring system

The SPH method is used to study the dynamic response of different mooring systems installed between floating platforms under the action of waves. It is compared with the dynamic response of offshore floating wind turbines with different mooring systems installed between floating platforms under the same environmental conditions to verify its damping performance. In order to make the numerical results more accurate, the simulation time of 20 s is divided into 500 steps. Two connection modes of mooring system between floating platforms are shown in Figure 4.



(a) Parallel Mooring System



(b) Cross Mooring System

Figure 4 Schematic diagram of floating platform intergroup mooring system

Analysis of numerical results

Parameter setting

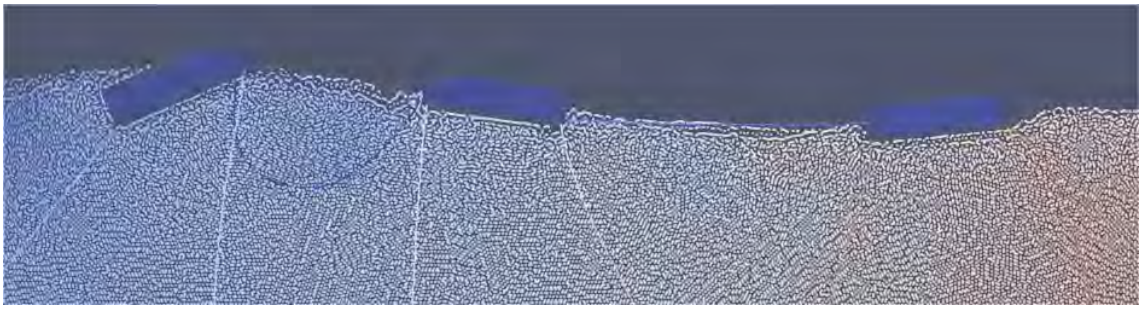
In this paper, two different mooring systems are used to explore the influence of different mooring systems on the stability of the floating platform, so as to improve its vibration reduction ability. The SPH code is used to simulate the floating platform groups of different mooring systems. The time integration scheme uses the Symplectic method, which is second-order accurate in time. It is an ideal choice for the Lagrangian scheme because it is time reversible and symmetrical without maintaining the geometric feature diffusion term [12]. The simulation parameters are shown in Table 1.

Table 1 Table of parameters

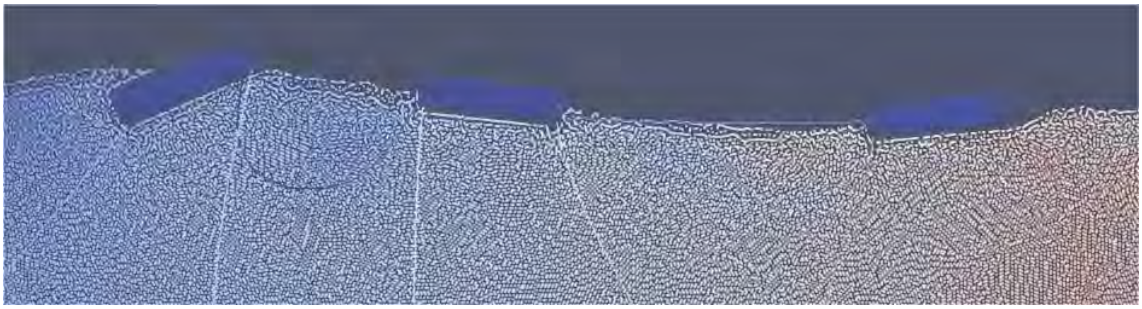
process	options
dimension	2D
domain (m)	26*5
kernel function	Wendland
time integration	Symplectic
viscosity	Artificial
boundary	DBC
simulation time (s)	20
initial particle distance (m)	0.02
h(m)	0.0367
fluid particles	111106
boundary particle	2674

Particle visualization of floating platform groups for different mooring systems

The simulation results show that after the floating platform is impacted by the simulated wave, we can observe the dynamic response of the floating platform under the parallel mooring system and the cross mooring system. Compared with the parallel mooring system, the cross mooring system performs better in reducing the motion amplitude of the platform, effectively reducing the displacement of the floating platform and achieving better vibration reduction effect. Observe the change of the floating platform position of the two mooring systems at the 300th step, and the particle schematics of the two mooring systems are shown in Figure 5.



(a) Parallel Mooring System



(b) Cross Mooring System

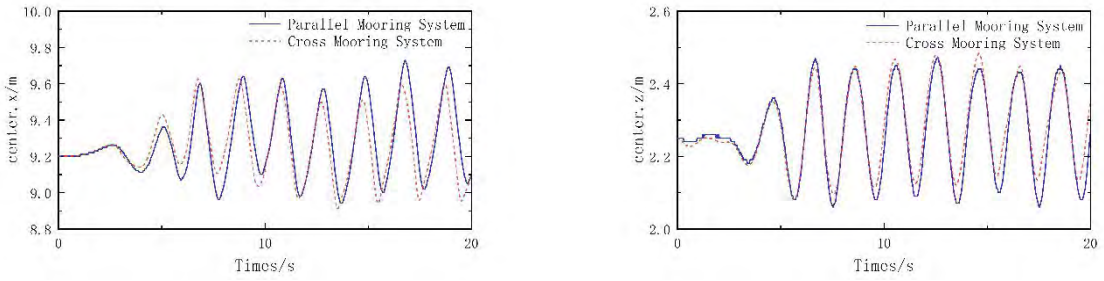
Figure 5 Illustration of the particle at the 300th step

The geometric center of the central floating platform changes

The influence of two different mooring systems on the geometric center of the floating platform under the impact of simulated waves. It can be seen from Table 2 and Figure 6 that the cross mooring system has an inhibitory effect on the geometric center of the floating platform on the x-axis and z-axis compared with the parallel mooring system, which further improves the stability and control performance of the platform.

Table 2 Main parameters of geometric center of floating platform (Unit: m)

Axis	Mooring System	Minimum value	Maximum value	Average value	Standard deviation
x-axis	**Parallel Mooring System**	8.94	9.73	9.28	0.193
	Cross Mooring System	8.91	9.63	9.26	0.176
z-axis	**Parallel Mooring System**	2.06	2.47	2.26	0.116
	Cross Mooring System	2.08	2.48	2.27	0.107



(a) **x-axis** (b) **z-axis**

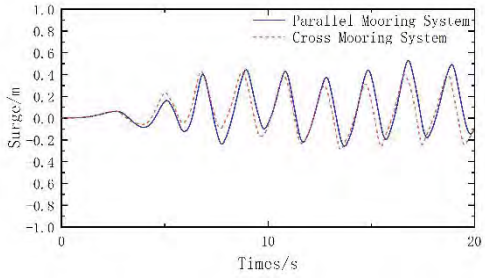
Figure 6 Schematic diagram of the geometric center change of the floating platform

The center floating platform changes with six degrees of freedom

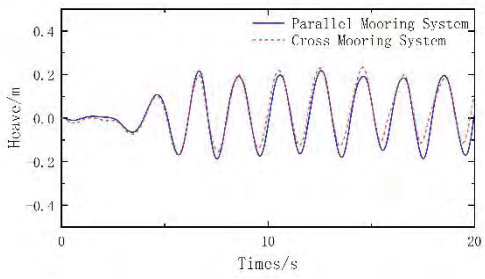
In this paper, the wave load generated by the wave is simulated to impact the floating platform group, resulting in the translation and rotation of the floating platform. The numerical model is 2D, so the main parameters on the six degrees of freedom of the floating platform include surge, heave and pitch. When studying the vibration amplitude of the floating platform, the root mean square (RMS) is used as the main parameter. The higher the root mean square value, the greater the vibration amplitude of the floating platform. By comparing the translational and rotational performance of the floating platform of different mooring systems, the vibration reduction effect on the floating platform is studied. The root mean square value provided in Table 3 can reflect the dynamic response of the central floating platform of two different mooring systems under wave load. Figure 7 shows the changes of surge, heave and pitch on the central floating platform.

Table 3 Center floating platform six degrees of freedom mean square error value

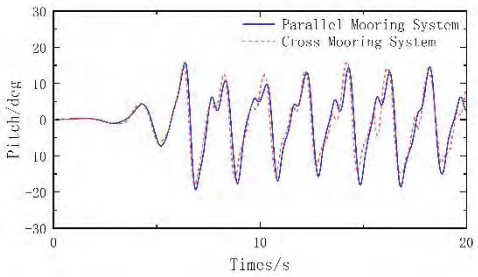
DOF	Parallel Mooring System	Cross Mooring System
Surge(m)	0.208	0.185
Heave(m)	0.117	0.110
Pitch(deg)	7.91	7.47



(a) Surge



(b) Heave



(c) Pitch

Figure 7 Schematic diagram of 6-DOF change of floating platform

The research shows that the vibration reduction ability of the cross mooring system is better than that of the parallel mooring system. In terms of translation, the vibration amplitude (root mean square) of Surge was reduced by 11.1 %, and the vibration amplitude (root mean square) of Heave was reduced by 5.98 %. In terms of rotation, the Pitch deflection amplitude (root mean square) was reduced by 5.56 %. On the whole, the cross mooring system is superior to the parallel mooring system in terms of translational and rotational stability. It has obvious advantages in vibration reduction and stability of the floating platform group, and provides more stable and reliable support for the operation of the floating platform group.

Conclusions

In this paper, the SPH method is used to simulate the floating platform groups of different mooring systems under simulated waves. The performance of different mooring systems in controlling the dynamic response of floating wind turbines is studied, especially in suppressing vibration. It overcomes the difficulty of traditional grid-based numerical methods in tracking the motion and deformation of solid objects and the free surface and fluid interface at the same time. Accurately simulate the dynamic response process of the floating platform group and make the simulation results more accurate. By comparing the parallel mooring system with the cross mooring system, the results show that the vibration reduction ability of

the cross mooring system will be more excellent, and the potential of the mooring system in reducing the motion of the wind turbine and improving the stability is verified, which provides a theoretical basis for the design of the mooring system structure of the floating wind turbine. In order to further improve the accuracy and practicality of the research, future work plans include more quantitative comparisons. In addition, future research will explore more factors that affect the dynamic response of wind turbine platforms, such as changes in the marine environment, wind speed and wave dynamics, so as to provide a more comprehensive theoretical and practical basis for the design and optimization of mooring systems for floating offshore wind turbines.

References

[1] R Liu and MB Liu. Smoothed particle hydrodynamics: A Meshfree Particle Method[M]. Spingapore: World Scientific 2003.

[2] Lucy L B. A numerical approach to the testing of the fission hypothesis. Astron J, 1977, 82: 1013–1024. https://doi.org/10.1086/112164.

[3] Gingold R A, Monaghan J J. Smoothed particle hydrodynamics: theory and application to non-spherical stars[J]. Monthly notices of the royal astronomical society, 1977, 181(3): 375-389. https://doi.org/10.1093/mnras/181.3.375.

[4] Tagliafierro B, Mancini S, Ropero-Giralda P, et al. Performance assessment of a planing hull using the smoothed particle hydrodynamics method[J]. Journal of Marine Science and Engineering, 2021, 9(3): 244. https://doi.org/10.3390/jmse9030244.

[5] Li S, Liu W K. Meshfree and particle methods and their applications[J]. Appl. Mech. Rev., 2002, 55(1): 1-34. https://doi.org/10.1115/1.1431547.

[6] Lind S J, Rogers B D, Stansby P K. Review of smoothed particle hydrodynamics: towards converged Lagrangian flow modelling[J]. Proceedings of the royal society A, 2020, 476(2241): 20190801. https://doi.org/10.1098/rspa.2019.0801.

[7] Capasso S, Tagliafierro B, Mancini S, et al. Regular wave seakeeping analysis of a planing hull by smoothed particle hydrodynamics: A Comprehensive Validation[J]. Journal of Marine Science and Engineering, 2023, 11(4): 700. https://doi.org/10.3390/jmse11040700.

[8] Yang Y, English A, Rogers B D, et al. Numerical modelling of a vertical cylinder with dynamic response in steep and breaking waves using smoothed particle hydrodynamics[J]. Journal of Fluids and Structures, 2024, 125: 104049. doi:10.1016/j.jfluidstructs.2023.104049.

[9] Tagliafierro B, Karimirad M, Martínez-Estévez I, et al. Numerical assessment of a tension-leg platform wind turbine in intermediate water using the smoothed particle hydrodynamics method[J]. Energies, 2022, 15(11): 3993. https://doi.org/10.3390/en15113993.

[10] Aslami M H, Rogers B D, Stansby P K, et al. Simulation of floating debris in SPH shallow water flow model with tsunami application[J]. Advances in Water Resources, 2023, 171: 104363. doi:10.1016/j.advwatres.2022.104363.

[11] Tagliafierro B, Karimirad M, Altomare C, et al. Numerical validations and investigation of a semi-submersible floating offshore wind turbine platform interacting with ocean waves using an SPH framework[J]. Applied Ocean Research, 2023, 141: 103757. doi:10.1016/j.apor.2023.103757.

[12] Liu J, Jiao P, Xu Y. Research on oil boom performance based on Smoothed Particle Hydrodynamics method[J]. Plos one, 2023, 18(7): e0289276. https://doi.org/10.1371/journal.pone.0289276.

Transient analysis of orthotropic thick and thin plates using low-order triangular elements

†J. Petrolito and D. Ionescu
Department of Engineering, La Trobe University, Australia.
†Corresponding author: j.petrolito@latrobe.edu.au

Abstract

This paper discusses the use of two alternative formulations, namely the free formulation and the deviatoric strain formulation, for developing low-order triangular finite elements that can be used for the dynamic analysis of orthotropic thick and thin plates. A key feature of both formulations is that they use interpolations for the transverse displacement and rotations that exactly satisfy the governing equations of Mindlin's theory. This ensures that the element approximations are consistent for both thick and thin plates, and no locking occurs in the thin plate limit. All the required element terms are developed in a straightforward manner in closed form. Newmark's method is used to solve the time-varying algebraic equations. Three examples are given in the paper to demonstrate the accuracy that is achieved with the proposed elements.

Keywords: Plate analysis, transient analysis, thick plates, shear deformation, finite elements.

Introduction

Plates are commonly-used components in structural engineering, for example as floor slabs, bridge decks and raft foundations. The behaviour of plates under transverse loading has been studied for many years. Classical plate theory was the first theory to be developed for this problem, and it still remains an important theory with a vast literature [1–3]. The theory considers the plate as a two-dimensional object with appropriate properties to resist the transverse loads. However, it ignores shear deformation, which is important for thicker plates. Alternative theories that include shear deformation have been developed in more recent years, and these theories aim to provide results that more closely agree with a three-dimensional analysis [2, 3].

Regardless of the theory that is used to describe the plate bending problem, exact solutions are only available for simple geometries and loads [1–3]. Hence, the analysis of practical plate problems that arise in structural engineering requires a numerical solution, and the finite element method is normally used [4, 5].

Originally, plate bending elements used classical thin plate theory as their basis. However, there were convergence and reliability issues in many cases, because of the difficultly of achieving C_1 continuity of the approximations [4]. More recently, plate theories that include shear deformation, such as Mindlin's first-order shear deformation theory [2, 6], have been preferred as a basis for plate bending elements. This is an attractive option as the theory is applicable to both thick and thin plates, and elements that use this theory as their basis only require C_0 continuity.

However, thick plate elements can also suffer from some problems. The two major problems for these elements are the possibility of unstable meshes due to spurious mechanisms, and lock-

ing of the solution in the case of thin plates. Many approaches have been used to alleviate these problems [4]. The locking problem can be avoided by using element interpolations that automatically satisfy the governing equations of Mindlin's theory, and several alternative formulations can be used to formulate thick plate elements using these interpolations [7, 8].

In this paper, the authors extend our previous work in this area [8] to develop triangular thick plate elements for the transient analysis of orthotropic plate structures. Transient analysis is important when the applied loads are time-varying, such as wind, earthquake and shock loads. The required element matrices can be derived in a straightforward manner in closed form. The resulting dynamics equations are solved using Newmark's method, which is unconditionally stable and can deal with both short- and long-term dynamic loads [4]. Three examples are given in the paper to demonstrate the accuracy that is achieved with the proposed elements.

Governing Equations

Mindlin's theory uses independent assumptions for the transverse displacement, w, and normal rotations, θ_x and θ_y (see Fig. 1).

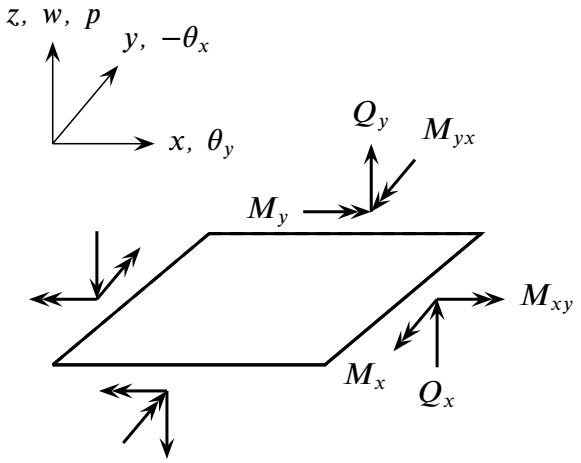


Figure 1. Sign convention.

Both bending and transverse shear strains are possible in this theory, and these are given by

$$\boldsymbol{\epsilon}_b = \begin{bmatrix} \partial/\partial x & 0 \\ 0 & \partial/\partial y \\ \partial/\partial y & \partial/\partial x \end{bmatrix} \begin{Bmatrix} \theta_x \\ \theta_y \end{Bmatrix} = \mathbf{L}_1 \boldsymbol{\theta} \tag{1a}$$

$$\boldsymbol{\epsilon}_s = \begin{Bmatrix} \partial/\partial x \\ \partial/\partial y \end{Bmatrix} w - \boldsymbol{\theta} = \mathbf{L}_2 w - \boldsymbol{\theta} \tag{1b}$$

where $\boldsymbol{\epsilon}_b$ is the bending strain vector and $\boldsymbol{\epsilon}_s$ is the shear strain vector, which is zero in the thin plate limit.

For an orthotropic material, bending moments and shear forces are given in terms of the strains

by

$$\mathbf{M} = \begin{Bmatrix} M_x \\ M_y \\ M_{xy} \end{Bmatrix} = \begin{bmatrix} D_x & D_1 & 0 \\ D_1 & D_y & 0 \\ 0 & 0 & D_{xy} \end{bmatrix} \boldsymbol{\epsilon}_b = \mathbf{E}_b \boldsymbol{\epsilon}_b \tag{2a}$$

$$\mathbf{Q} = \begin{Bmatrix} Q_x \\ Q_y \end{Bmatrix} = kh \begin{bmatrix} G_{xz} & 0 \\ 0 & G_{yz} \end{bmatrix} \boldsymbol{\epsilon}_s = \mathbf{E}_s \boldsymbol{\epsilon}_s \tag{2b}$$

with $M_{yx} = M_{xy}$. In Eq. (2), D_x, D_y, D_1 and D_{xy} are bending rigidities, G_{xz} and G_{yz} are transverse shear moduli, h is the plate thickness and k is the shear correction factor, which is usually taken as $5/6$ or $\pi^2/12$.

The terms in $\mathbf{E}_b$ are given in terms of the fundamental material properties by

$$D_x = \frac{E_x h^3}{12(1 - \nu_{xy}\nu_{yx})}, \; D_y = \frac{E_y h^3}{12(1 - \nu_{xy}\nu_{yx})}, \; D_1 = \frac{\nu_{xy} E_y h^3}{12(1 - \nu_{xy}\nu_{yx})}, \; D_{xy} = \frac{G_{xy} h^3}{12} \tag{3}$$

where E_x and E_y are Young's moduli, ν_{xy} and $\nu_{yx} = \nu_{xy} E_y / E_x$ are Poisson's ratios and G_{xy} is the in-plane shear modulus. For an isotropic material, $D_x = D_y = D = Eh^3/[12(1 - \nu^2)]$, $D_1 = \nu D$, $D_{xy} = D(1 - \nu)/2$ and $G_{xy} = G_{xz} = G_{yz} = E/[2(1 + \nu)]$.

The dynamic equilibrium equations are

$$\mathbf{L}_3 \boldsymbol{\sigma} + \mathbf{p} = \mathbf{p}_I \tag{4}$$

where

$$\boldsymbol{\sigma} = \begin{Bmatrix} \mathbf{M} \\ \mathbf{Q} \end{Bmatrix}, \quad \mathbf{p} = \begin{Bmatrix} \mathbf{0} \\ p \end{Bmatrix}, \quad \mathbf{p}_I = \rho h \begin{Bmatrix} \frac{h^2}{12} \ddot{\boldsymbol{\theta}} \\ \ddot{w} \end{Bmatrix}, \quad \mathbf{L}_3 = \begin{bmatrix} \mathbf{L}_1^T & \mathbf{I} \\ \mathbf{0} & \mathbf{L}_2^T \end{bmatrix} \tag{5}$$

In Eq. (5), p is the time-varying applied transverse load, ρ is the material density, a dot denotes partial differentiation with respect to time, t, $\mathbf{0}$ represents a zero matrix and $\mathbf{I}$ represents an identity matrix. The terms in $\mathbf{p}_I$ in Eq. (5) that are associated with the rotations are frequently neglected [2].

Finite Element Formulation

Exact solutions of the governing equations of Mindlin's theory are only available for simple problems, and a numerical approach is usually required to analyse practical plate structures. While many numerical methods are available, the most general and flexible method is the finite element method [4].

In the finite method element, the continuous problem is discretised by splitting the plate region of interest into elements, usually triangular or quadrilaterals elements, and approximating the plate displacement and rotations over each element with specified functions in terms of nodal variables that need to be calculated. In the most common formulation [4], this process approximates the governing partial differential equations in Eq. (4) with a set of time-varying algebraic equations, namely

$$\mathbf{K}\mathbf{x} + \mathbf{M}\ddot{\mathbf{x}} + \mathbf{C}\dot{\mathbf{x}} = \mathbf{R} \tag{6}$$

where **K** is the global stiffness matrix, **M** is the global mass matrix, **C** is the global damping matrix, **R** is the global load vector and **x** is the global vector of unknown variables.

The solution of Eq. (6) usually requires a numerical technique, and many such techniques have been developed and continue to be developed. In practice, numerical time-stepping methods are generally preferred as they are flexible, and combine efficiency and robustness, particularly for large problems [4, 9–12]. In this paper, we have used Newmark's time-stepping method, which is unconditionally stable with its default parameters [4]. Hence, it has been found to be a reliable method for solving dynamics problems.

Element Approximations and Formulations

Fig. 2 shows the geometry and degrees of freedom for the two elements considered in this paper. They have three degrees of freedom at each node, making a total of nine degrees of freedom. Element TF9 is based on the free formulation [13], while element TD9 is based on the extension of the deviatoric strain formulation to thick plate problems [8, 14]. Full details of the derivation of the stiffness matrices were given previously by the authors [8]. Hence, we summarise this development here, and in addition consider the extension of our previous work to dynamics problems. It should be noted that the general methodology described below can also be used to develop higher-order triangular elements, rectangular elements and quadrilateral elements. However, we do not consider such elements in this paper for brevity.

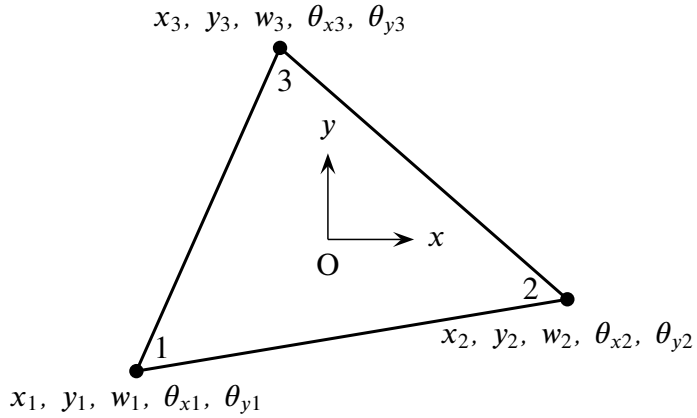


Figure 2. Triangular element (origin is at centre of element).

A fundamental feature of the elements considered in this paper is that their approximations satisfy the governing differential equations for Mindlin's theory. Accordingly, the approximations for w, θ_x and θ_y are taken as

$$\mathbf{u} = \begin{Bmatrix} \theta \\ w \end{Bmatrix} = \mathbf{N}\mathbf{v} \tag{7}$$

where **N** is the shape function matrix and

$$\mathbf{v} = \left[w_1,\ \theta_{x1},\ \theta_{y1},\ w_2,\ \theta_{x2},\ \theta_{y2},\ w_3,\ \theta_{x3},\ \theta_{y3}\right] \tag{8}$$

is the vector of the degrees of freedom for the element. All the terms in **N** are homogenous solutions of Eq. (4) with $\mathbf{p} = \mathbf{p}_I = \mathbf{0}$ [8]. After inter-element continuity is enforced and all the elements are assembled, the assembled **v** vectors for all elements become **x** in Eq. (6).

Both formulations assume that the element stiffness matrix, $\mathbf{k}$, can be expressed as

$$\mathbf{k} = \mathbf{k}_{\text{basic}} + S\mathbf{k}_{\text{high}} \tag{9}$$

where $\mathbf{k}_{\text{basic}}$ is the basic stiffness matrix, $\mathbf{k}_{\text{high}}$ is the higher-order stiffness matrix and S is a positive scale factor [8, 15]. Using Eqs. (7) and (9), the element stiffness matrix for both elements can be formulated in a straightforward manner using analytical integration [8].

The element mass matrix, $\mathbf{m}$, is required for dynamics problems, and it is given by

$$\mathbf{m} = \rho h \int_A \left[\mathbf{N}_w^T \mathbf{N}_w + \frac{h^2}{12} \mathbf{N}_\theta^T \mathbf{N}_\theta \right] dA \tag{10}$$

where $\mathbf{N}_w$ and $\mathbf{N}_\theta$ are the displacement and rotation submatrices of $\mathbf{N}$, respectively, in Eq. (7) and A is the area of the element. Eq. (10) gives the consistent mass matrix, which is fully populated. An alternative is to use a lumped mass approach, which leads to a diagonal mass matrix. While this choice is computationally more efficient, it may be less accurate than the consistent mass matrix [4].

The consistent element load vector, $\mathbf{r}$, is given by

$$\mathbf{r} = \int_A p\mathbf{N}_w^T \, dA \tag{11}$$

Again, an alternative to Eq. (11) is to use a lumped load vector [4].

The global stiffness and mass matrices and the load vector in Eq. (6) are assembled from the corresponding element terms in Eqs. (9) to (11) using standard finite element procedures [4]. However, the global damping matrix, $\mathbf{C}$, in Eq. (6) generally requires a different treatment.

While it is possible to formulate an element damping matrix using the element approximations in Eq. (7), this is usually not done because it is difficult to quantify damping accurately at an element level. Hence, the preferred option for the finite element method is to approximate $\mathbf{C}$ from $\mathbf{K}$ and $\mathbf{M}$, and this is the method used in this paper. Accordingly, $\mathbf{C}$ is approximated by the Rayleigh damping formulation [4], and is given by

$$\mathbf{C} = a_0\mathbf{M} + a_1\mathbf{K} \tag{12}$$

where the constants a_0 and a_1 are chosen to give particular values of damping for two specified natural frequencies of the structure.

Examples

We analyse three benchmark examples to illustrate the behaviour of the proposed elements for transient analysis. Each example considers a simply supported square plate. For Mindlin's theory, the appropriate boundary conditions for a simple support are $w = \theta_t = M_{nn} = 0$, where n and t are the normal and tangential directions, respectively, at the boundary. The scale factor was taken as $S = 1$, the shear correction factor was taken as $k = 5/6$, and a consistent mass matrix and load vector were used.

In general, there are two sources of errors for transient analysis using finite elements. The first error is the usual approximation error that is inherent with finite elements [4]. The second error

is due to solving Eq. (6) using a particular value of the time step [9]. Hence, convergence requires both a finer mesh and a smaller time step to be consistently used. A given mesh for an example will be considered acceptable if the results for the mesh are within the typical engineering accuracy of 2% - 5% relative to the target value.

Simply supported thin plate

Fig. 3 shows a square plate size $a \times a$ and thickness h that is subjected to a suddenly applied uniform load p. For each example we consider in this paper, we assume that the properties and loads are symmetric so that only a quarter of the plate needs to be analysed as shown in the figure.

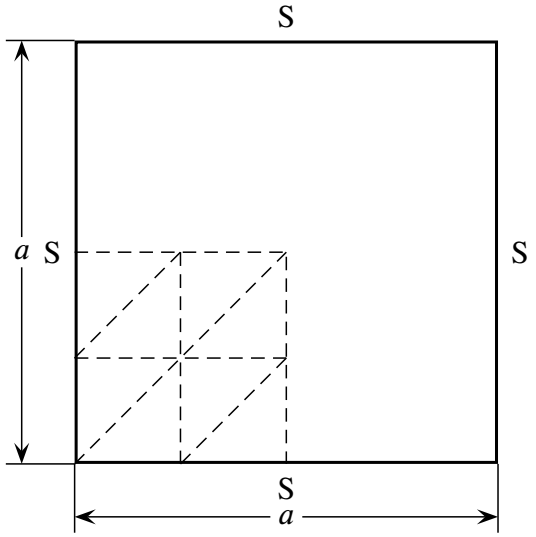


Figure 3. Square plate of size $a \times a$ with typical mesh for quarter section (S = simple support; $N = 2$).

As a first example, we consider a simply supported thin plate with parameters that correspond to the NAFEMS benchmark test T13 [16]. Accordingly, we take $a = 10$ m, $h = 0.05$ m and $p = 100$ Pa. The span to thickness ratio of 200 corresponds to a thin plate, and shear deformation is small for this example. The material is isotropic with $E = 200$ GPa, $\nu = 0.3$ and $\rho = 8000$ kg/m^3. A damping ratio of 2% of the critical damping for the first natural frequency was used, and this corresponds to damping constants $a_0 = 0.299$/sec and $a_1 = 1.339 \times 10^{-3}$ sec. The specified time step of $\Delta t = 0.002$ sec [16] was used in Newmark's method.

Taking symmetry into account, a quarter of the plate was analysed using a uniform mesh of $2N \times 2N$ triangular elements. Results for the peak central displacement, w_p, the corresponding time for the peak central displacement, t_p, and the peak central stress, σ_{xp}, where $\sigma_x = 6M_x/h^2$, are given in Table 1. The results are compared with the reference solutions in [16, 17].

Table 1. Peak displacement, w_p (mm), time for peak displacement, t_p (sec), and peak stress, σ_{xp} (MPa), at centre of simply supported thin plate subjected to a suddenly applied uniform load.

	TF9			TD9		
N	w_p	t_p	σ_{xp}	w_p	t_p	σ_{xp}
2	3.178	0.198	2.315	3.672	0.208	3.083
4	3.416	0.216	2.408	3.538	0.214	2.701
8	3.475	0.210	2.423	3.506	0.212	2.517
16	3.490	0.210	2.423	3.498	0.212	2.454
32	3.494	0.210	2.429	3.496	0.212	2.437
Ref. [16]	3.523	0.210	2.484	3.523	0.210	2.484
Ref. [17]	3.447	0.215	2.236	3.447	0.215	2.236

The peak displacements for both elements converge rapidly to values that are acceptably close to the reference values, with $N \geq 4$ being an acceptable mesh for the displacement. As expected, the peak stress converges more slowly, with $N \geq 8$ being an acceptable mesh for the stress. The differences between the results for both elements are small as N increases. Figs. 4 and 5 show the variations in time for the central displacement and stress for element TF9. These figures provide a visual confirmation of the convergence of the results. The results for element TD9 are similar.

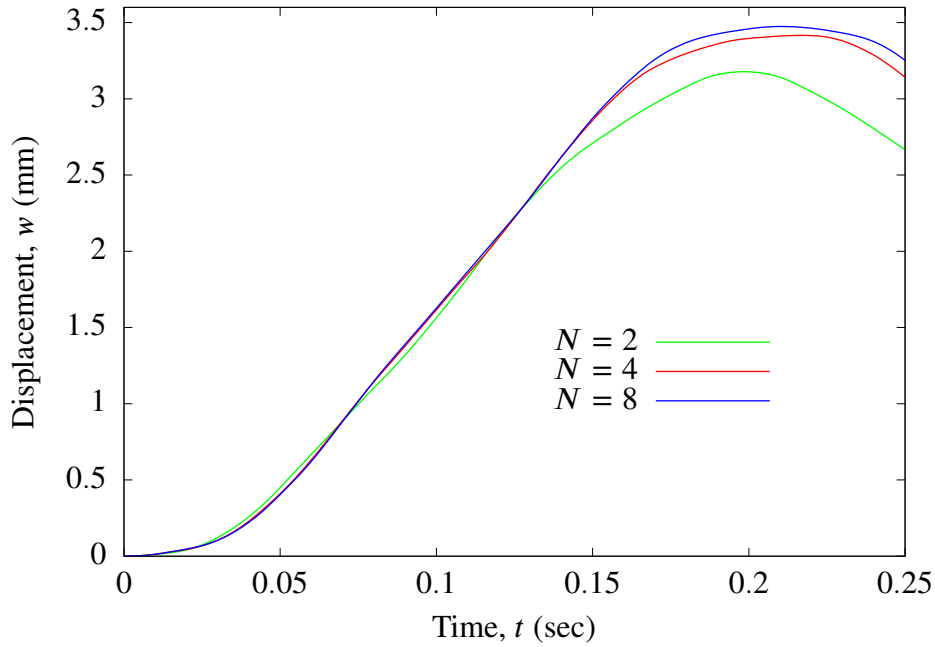


Figure 4. Variation of central displacement with time for simply supported thin plate subjected to suddenly applied uniform load (element TF9).

Simply supported thick plate

The plate in Fig. 3 is now analysed using a thickness of $t = 1$ m and $p = 1$ MPa. The span to thickness ratio of 10 corresponds to a thick plate, and shear deformation is significant for

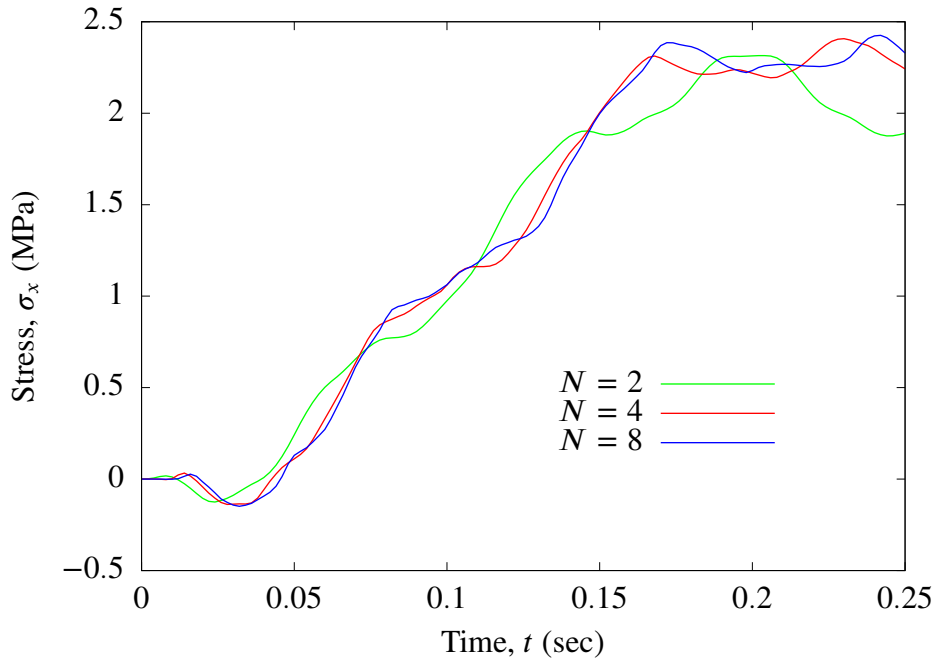


Figure 5. Variation of central stress with time for simply supported thin plate subjected to suddenly applied uniform load (element TF9).

this example. The material properties are the same as for the previous example. The damping constants were taken as $a_0 = 5.772$/sec and $a_1 = 6.929 \times 10^{-5}$ sec, which again correspond to a damping ratio of 2% of the critical damping for the first natural frequency. A time step of $\Delta t = 0.0001$ sec was used. This example corresponds to the NAFEMS benchmark test T20 [16].

The numerical results in Table 2 show that $N \geq 8$ gives acceptable accuracy for the peak displacement and stress for both elements. Figs. 6 and 7 provide a visual confirmation of the convergence of the results for element TF9, and the results for element TD9 show a similar trend.

Table 2. Peak displacement, w (mm), time for peak displacement, t_p (sec), and peak stress, σ_{xp} (MPa), at centre of simply supported thick plate subjected to suddenly applied uniform load.

	TF9			TD9		
N	w_p	t_p	σ_{xp}	w_p	t_p	σ_{xp}
2	4.043	0.0104	55.66	4.782	0.0108	72.61
4	4.416	0.0104	59.74	4.643	0.0106	63.52
8	4.546	0.0106	60.77	4.612	0.0106	61.19
16	4.587	0.0106	60.62	4.605	0.0106	60.62
32	4.598	0.0106	60.50	4.603	0.0106	60.49
Ref. [16]	4.524	0.0108	62.11	4.524	0.0108	62.11
Ref. [17]	4.612	0.0110	57.67	4.612	0.0110	57.67

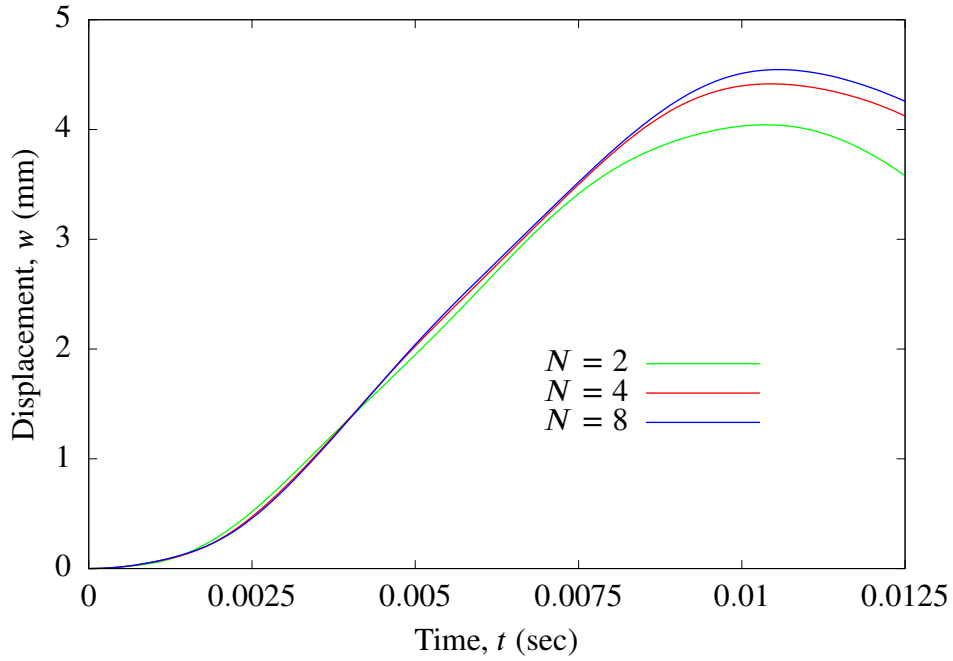


Figure 6. Variation of central displacement with time for simply supported thick plate subjected to suddenly applied uniform load (element TF9).

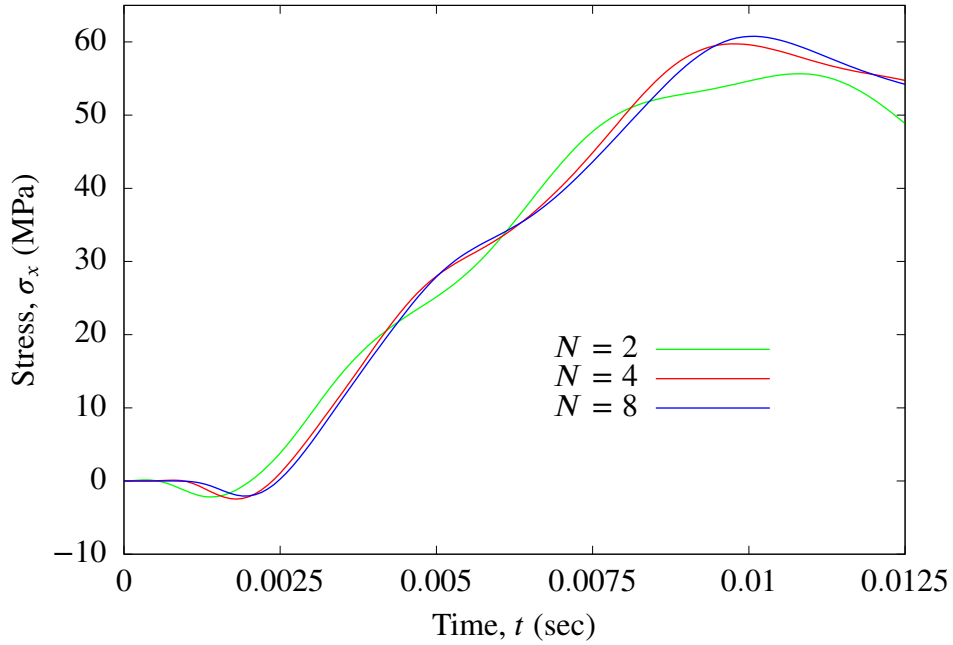


Figure 7. Variation of central stress with time for simply supported thick plate subjected to suddenly applied uniform load (element TF9).

Simply supported orthotropic thick plate

As a final example, we consider the plate shown in Fig. 3 with the same geometry as the second example ($a = 10$ m and $h = 1$ m), but with orthotropic properties. In this case, the material properties were taken as $E_y = 9$ GPa, $E_x = 15E_y$, $\nu_{xy} = 0.3$, $G_{xy} = E_y$, $G_{xz} = E_y$ and $G_{yz} = 0.9E_y$. The uniform load was taken as $p = 0.5$ MPa. The damping constants were taken

as $a_0 = 92$/sec and $a_1 = 2.7 \times 10^{-5}$ sec, which correspond to a damping ratio of approximately 5% of the critical damping for the first natural frequency. As for the second example, a time step of $\Delta t = 0.002$ sec was used for the analysis.

The results in Table 3 again show that both elements converge rapidly, and a value of $N \geq 8$ is an acceptable mesh for the problem. Figs. 8 and 9 also provide a visual confirmation of the convergence of the results for element TF9, and the results for element TD9 are again similar.

Table 3. Peak displacement, w (mm), time for peak displacement, t_p (sec), and peak stress, σ_{xp} (MPa), at centre of simply supported orthotropic thick plate subjected to suddenly applied uniform load.

	TF9			TD9		
N	w_p	t_p	σ_{xp}	w_p	t_p	σ_{xp}
2	10.47	0.0099	64.51	11.52	0.0107	72.55
4	11.36	0.0102	67.20	11.82	0.0104	70.19
8	11.77	0.0104	70.15	11.94	0.0105	70.67
16	11.90	0.0105	70.20	11.97	0.0105	70.36
32	11.96	0.0104	70.21	11.99	0.0104	70.14

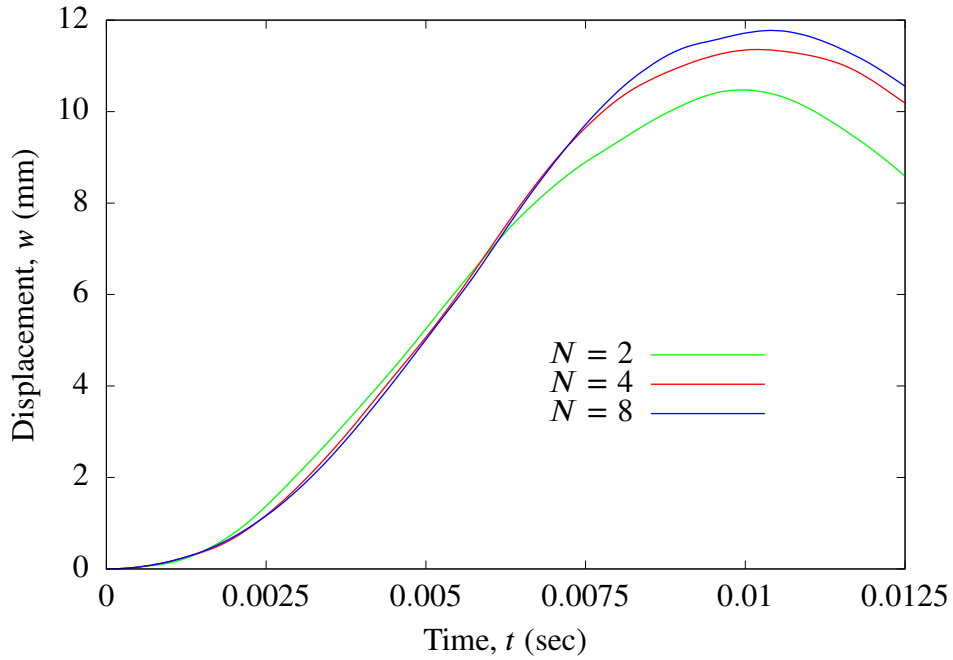


Figure 8. Variation of central displacement with time for simply supported orthotropic thick plate subjected to suddenly applied uniform load (element TF9).

Conclusions

This paper has discussed the development of two, low-order triangular elements for the dynamic analysis of orthotropic thick and thin plate structures. The key feature of the elements is that they use element interpolations that are the solutions of the governing equations of Mindlin's theory. Hence, locking cannot occur in the thin plate limit with these elements. Both formulations lead to reliable and accurate elements. Three examples were considered in the paper to demonstrate the accuracy that is achieved with the proposed elements.

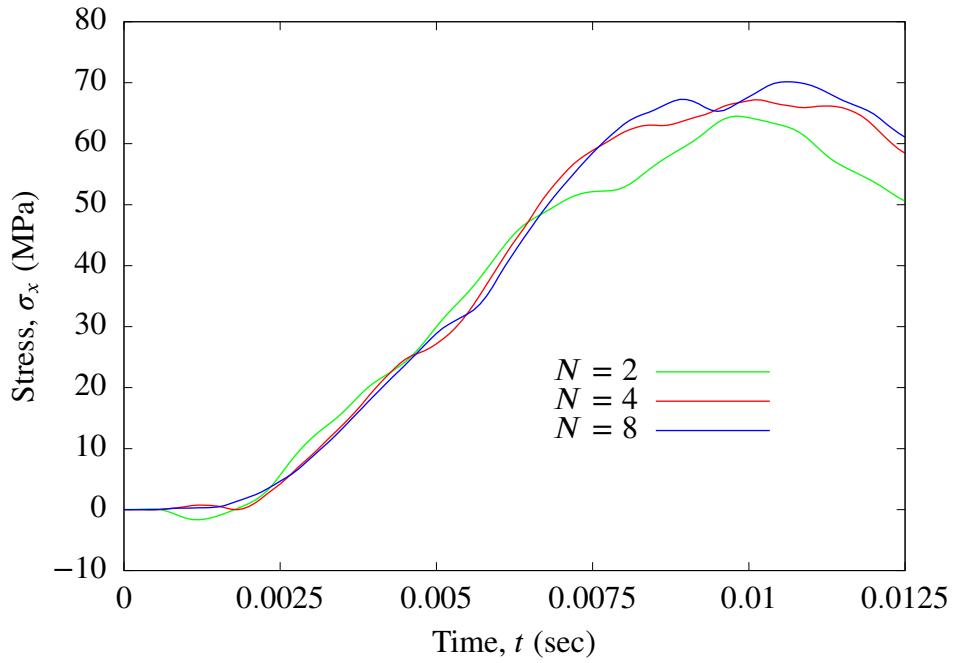


Figure 9. Variation of central stress with time for simply supported orthotropic thick plate subjected to suddenly applied uniform load (element TF9).

References

[1] Timoshenko, S. and Woinowsky-Krieger, S. (1959) *Theory of Plates and Shells*, 2nd edn., McGraw-Hill, New York.

[2] Reddy, J. (2007) *Theory and Analysis of Elastic Plates and Shells*, 2nd edn., Taylor and Francis, Philadelphia.

[3] Mittelstedt, C. (2023) *Theory of Plates and Shells*, Springer-Verlag, Berlin.

[4] Zienkiewicz, O., Taylor, R. and Fox, D. (2014) *The Finite Element Method for Solid and Structural Mechanics*, vol. 2, 7th edn., Butterworth-Heinemann, Oxford.

[5] Werkle, H. (2021) *Finite Elements in Structural Analysis*, Springer Nature, Cham.

[6] Mindlin, R. (1951) Influence of rotatory inertia and shear on flexural motion of isotropic plates, *Journal of Applied Mechanics*, **23**, 31–38.

[7] Petrolito, J. (2014) Vibration and stability analysis of thick orthotropic plates using hybrid-Trefftz elements, *Applied Mathematical Modelling*, **38**, 5858–5869.

[8] Petrolito, J. and Ionescu, D. (2022) Alternative formulations for finite elements for thick plate analysis, in G. Liu and N. Hung (editors), *Proceedings, 13th International Conference on Computational Methods*, 1–12, ScienTech Publisher.

[9] Wood, W. (1990) *Practical Time-Stepping Schemes*, Oxford University Press, Oxford.

[10] Kim, W. and Reddy, J. (2020) Novel explicit time integration schemes for efficient transient analyses of structural problems, *International Journal of Mechanical Sciences*, **172**, 10.1016/j.ijmecsci.2020.105429.

[11] Liu, T., Wang, P. and Wen, W. (2023) A novel single-step explicit time integration method based on momentum corrector technique for structural dynamic analysis, *Applied Mathematical Modelling*, **124**, 1–23.

[12] Kim, W. (2024) Improved two-stage implicit time integration methods with unconventionally determined parameters for analyses of linear and nonlinear structural dynamics, *Nonlinear Dynamics*, **112**, 1221–1259.

[13] Bergan, P. and Nygård, M. (1984) Finite elements with increased freedom in choosing shape functions, *International Journal for Numerical Methods in Engineering*, **20**, 643–663.

[14] Militello, C. and Felippa, C. (1991) The first ANDES elements: 9-dof plate bending triangles, *Computer Methods in Applied Mechanics and Engineering*, **93**, 217–246.

[15] Bergan, P. and Felippa, C. (1985) A triangular membrane element with rotational degrees of freedom, *Computer Methods in Applied Mechanics and Engineering*, **50**, 25–69.

[16] Maguire, J., Dawswell, D. and Gould, L. (1989) *Selected Benchmarks for Forced Vibration*, NAFEMS, Glasgow.

[17] Abaqus (2015) *Abaqus Benchmarks Guide*, Dassault Systèmes, Boston.

AUTHOR INDEX